DOCUMENT RESUME

ED 098 905 HE 006 087

AUTHOR	Weldon, Kent; Huckfeldt, Vaughn
TITLE	Prototype Software for a Federal Planning Model for Analysis of Accessibility to Higher Education.
INSTITUTION	Western Interstate Commission for Higher Education, Boulder, Colo. National Center for Higher Education Management Systems.
SPONS AGENCY	Office of Education (DHEW), Washington, D.C.
PUB DATE	73
CONTRACT	OEC-0-72-3575
NOTE	98p.; Related documents are HE 006 086 and 088
EDRS PRICE	MF-$0.75 HC-$4.20 PLUS POSTAGE
DESCRIPTORS	*Computer Programs; Educational Demand; *Educational Planning; Educational Supply; Federal Aid; *Federal Programs; Financial Support; *Higher Education; Management Systems; *Models; Private Financial Support; State Aid

ABSTRACT

This report presents the prototype software resulting from an initial research effort in the development of a comprehensive national planning model for higher education. The software for the prototype model requires as input existing or derivable institutional and student data and is designed to permit prototype planning studies to examine the impact of alternative federal programs on accessibility and, to a lesser extent, on institutional viability. The model also will assist in identifying high-payoff areas for future research on a comprehensive planning model and additional data requirements of such a model. The model software is written in ANS FORTRAN for use on a C.D.C. 6400 computer and requires 30K 60-bit words and utilizes two 50K random access data storage files. The design provides for interactive or batch modifications to the data files, remote batch processing with interrogation summary reports, and batch listing of detailed output information. (Author)

ED 098905

PROTOTYPE SOFTWARE FOR A FEDERAL PLANNING MODEL FOR ANALYSIS OF ACCESSIBILITY TO HIGHER EDUCATION

Kent Weldon

Vaughn Huckfeldt

1973

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

This report is part of a research program supported by the Department of Health, Education, and Welfare Contract No. OEC-0-72-3575. Ideas and opinions expressed in this paper are those of the authors and do not necessarily reflect an official position of NCHEMS, WICHE, or the U.S. Office of Education.

HE 006 087

NATIONAL CENTER FOR HIGHER EDUCATION MANAGEMENT SYSTEMS AT
WESTERN INTERSTATE COMMISSION FOR HIGHER EDUCATION
P. O. Drawer P Boulder, Colorado 80302

An Equal Opportunity Employer

ABSTRACT

This report presents the prototype software resulting from an initial research effort in the development of a comprehensive national planning model for higher education. The software for the prototype model requires as input existing or derivable institutional and student data and is designed to permit prototype planning studies to examine the impact of alternative federal programs on accessibility and, to a lesser extent, on institutional viability. The model also will assist in identifying high-payoff areas for future research on a comprehensive planning model and additional data requirements of such a model.

The model software is written in ANS FORTRAN for use on a C.D.C. 6400 computer and requires 30K 60 bit words and utilizes two 50K random access data storage files. The design provides for interactive or batch modifications to the data files, remote batch processing with interrogation summary reports, and batch listing of detailed output information. The data structure takes advantage of the efficiencies of link list structures in both storage and processing.

The software described in this document consists of:

a. MODIFY -- a routine that creates or updates the data base for the prototype model.

b. NPM 2.4 -- the current version of the prototype model.

c. VIEW -- output report routine that displays several summary reports from runs of the model comparing two alternative financing plans.

PREFACE

The preliminary software for the federal model for analysis of accessibility to higher education has been programmed and documented in this report by Kent Weldon and Vaughn Huckfeldt of the NCHEMS staff. The preliminary software and report have not been reviewed or approved by members of the NCHEMS advisory structure as of this printing.

This report is released to provide a comprehensive review of the software for the prototype model. The software for the model in its current state of development will permit prototype planning studies and research on the model capabilities, and will assist in the identification of high-payoff areas of research needed for a more comprehensive planning model. The software for the model is designed for use by the Office of Planning, Budgeting, and Evaluation in the U.S. Office of Education, and research agencies interested in planning models for higher education. The prototype model software should not be used without due consideration given to its design assumptions, limitations on data reliability, and the fact that the model has not been pilot-tested. Consideration should also be given to the fact that the prototype model is not a comprehensive national planning model, since it provides only for analysis of accessibility to higher education. This report and

the prototype model software are being released as Type II NCHEMS software (not supported or guaranteed) and, depending on additional funding, may be replaced within one year by improved prototype software that has been fully pilot-tested. The model describes terms necessary for national planning but does not attempt to set national standards on these terms.

TABLE OF CONTENTS

Chapter		Page
I.	Introduction	1
II.	List of Variables	3
III.	NPM 2.4	6
	Program NPM	6
	Subroutine MAYNE	17
	Subroutine NEWX	23
	Subroutine CRITER	26
	Subroutine UPTRAN	29
	Subroutine DEBUG	33
	Subroutine UXDUMP	35
	Subroutine CONTROL	36
	Subroutine VIEWGO	38
	Subroutine DATA	41
	Block DATA	45
	Subroutine DSINV	46
	Subroutine DMFSD	49
	Subroutine CALCV	51
	Subroutine MDUMP	61
	Subroutine VDUMP	62
	Function LINEQU	63
IV.	MODIFY	65
	Program MODIFY	65
	Subroutine MOD1	67
	Subroutine MOD2	68
	Subroutine MOD3	69
	Subroutine MOD4	70
	Subroutine MOD5	71
	Subroutine MOD6	72
	Subroutine DATA	73
	Subroutine S	74
	Subroutine QT	75
	Subroutine T	76
	Subroutine SPEC	77

TABLE OF CONTENTS (Cont.)

Chapter		Page
V.	VIEW	78
	PROGRAM VIEW.	78
	Subroutine SUBA	83
	Subroutine SUBI	85
VI.	Description of Other Related Reports	87

CHAPTER I

INTRODUCTION

Under a National Planning Model project contract from USOE, NCHEMS has developed a prototype higher education model for the analysis of the impact of alternative financing plans on accessibility to higher education and institutional viability (the ability of institutions to survive with a given level of federal aid). The basic purpose of the National Planning Model project is to provide an initial prototype higher education model that would permit prototype planning studies to demonstrate the feasibility of such analytical tools for a national policy analysis of higher education and that would also assist in identifying high-payoff areas of research needed to develop a comprehensive national planning model.

The main sections of this document contain a complete listing of the software for the prototype federal planning model for analysis of accessibility to higher education. A list of the variables is presented to provide a reference to be used with the software listings, then the program listings follow, and the report concludes with a description of other related reports.

Each program contains liberal comments to describe the purpose of that particular subroutine or of a certain subsection or code. This should not be viewed as complete software documentation, since the report does not contain a general systems flow discussion and specific flow discussions of each routine. More comprehensive documentation would be part of complete user's manual, which may be developed in the future, depending upon additional project funding.

CHAPTER II

LIST OF VARIABLES

Control Variables	Units	(Matrix U)
$u_1 - u_4$	(people)	Faculty (Full instr.) hired
u_5	(ASF)	Instructional space begun
u_6	(ASF)	Noninstructional space begun
$u_7 - u_{22}$	(people)	Lower-division students admitted
u_{23}	(people)	Grad students admitted
u_{24}	(dollars)	Gifts spent on construction
u_{25}	(dollars)	Gifts spent on operations
u_{26}	(dollars)	Capital financed
$u_{27} - u_{37}$	(dollars)	Institutional aid to students by (A, I, L) ability, income, and level
u_{38}	(dollars)	Tuition

State Variables	Units	(Matrix X)
$x_1 - x_4$	(people)	Number of faculty
$x_5 - x_7$	(ASF)	Instructional space under construction (t-3, t-2, t-1)
x_8	(ASF)	Current instructional space
$x_9 - x_{11}$	(ASF)	Noninstructional space under construction
x_{12}	(ASF)	Current noninstructional space

State Variables	Units	(Matrix X)
$x_{13} - x_{28}$	(people)	Lower-division students
$x_{29} - x_{44}$	(people)	Upper-division students
x_{45}	(people)	Graduate students
x_{46}	(dollars)	Current restricted endowment
x_{47}	(dollars)	Current unrestricted endowment
x_{48}	(dollars)	Net cash balance (construction)
x_{49}	(dollars)	Net cash balance (operating)

Exogeneous Variables (Matrix Z) - Sector-dependent, time-dependent

$z_1 - z_{16}$	(people)	Transfer students (I, A)
z_{17}	(dollars)	Restricted gifts
z_{18}	(dollars)	Unrestricted gifts
z_{19}	(dollars)	State construction funds (LUMP)
z_{20}	(dollars)	Federal construction funds (LUMP)
z_{21}	(dollars)	State operating funds (LUMP)
z_{22}	(dollars)	Federal operating funds (LUMP)

Multi-Sector Variables (Matrix VMS) - Sector-dependent, time-dependent

$v_1 - v_{33}$	(dollars)	Federal loans to students
$v_{34} - v_{66}$	(dollars)	Federal grants to students
v_{67}	(%)	Value of loans in grand dollars

Exogeneous Transition Variables (Matrix Q) - Sector-dependent, time-dependent

q_1	(dollars/ASF)	State construction funds
a_2	(dollars/student)	Funds (constr)/LD student
q_3	(dollars/student)	Funds (constr)/UD student
q_4	(dollars/student)	Funds (constr)/Grad student
q_5	(dollars/ASF)	Federal constructions funds/ASF
q_6	(dollars/student)	Construction funds/LD
q_7	(dollars/student)	Construction funds/UD
q_8	(dollars/student)	Construction funds/Grad
q_9	(dollars/student)	State operating funds/student
q_{10}	(dollars/student)	Administrative costs/student
$q_{11} - q_{42}$	(dollars/student)	Federal aid to inst/student (I, A, L)
q_{43}	(dollars/student)	Federal aid to inst/Grad

BEST COPY AVAILABLE

CHAPTER III


```
      PROGRAM NPM (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE42,TAPE1,TANPM-0010
     1PE2)                                                              NPM-0020
C                                                                       NPM-0030
C          NATIONAL PLANNING MODEL--VERSION 2.4 (9/10/72)               NPM-0040
C                                                                       NPM-0050
C     --PREPARED FOR UNITED STATES OFFICE OF EDUCATION BY THE           NPM-0060
C     --NATIONAL CENTER FOR HIGHER EDUCATION MANAGEMENT SYSTEMS         NPM-0070
C     --AT WICHE UNDER USOE CONTRACT-                                   NPM-0080
C                                                                       NPM-0090
C                          OEC-0-72-3575                                NPM-0100
C                                                                       NPM-0110
C                                                                       NPM-0120
C     AUTHORS-G.WEATHERSBY,V.HUCKFELDT,W.KIRSCHLING                     NPM-0130
C     IMPLEMENTOR-K.WELDON                                              NPM-0140
C                                                                       NPM-0150
C                                                                       NPM-0160
C                                                                       NPM-0170
C        ***NOTE***      NO WARRANTY, EITHER EXPRESS OR IMPLIED,        NPM-0180
C                        IS MADE FOR THIS COLLECTION OF PROGRAMS        NPM-0190
C                        OR FOR ITS ACCOMPANYING TEST DATA.             NPM-0200
C                        ALL CURRENT SOFTWARE FOR THIS PROJECT IS       NPM-0210
C                        RELEASED AS NCHEMS TYPE II SOFTWARE,           NPM-0220
C                        WITH NO PROVISION MADE FOR PROGRAM ACCURACY    NPM-0230
C                        OR MAINTENANCE.                                NPM-0240
C                                                                       NPM-0250
C                                                                       NPM-0260
C                                                                       NPM-0270
C     THIS PROGRAM MAY USE A VARYING NUMBER OF INPUT AND OUTPUT DATA    NPM-0280
C     STREAMS,DEPENDING ON THE CAPABILITIES AND ACCESSING DIFFICULTIES  NPM-0290
C     FOR A PARTICULAR INSTALLATION.  THE MINIMUM CONFIGURATION REQUIRESNPM-0300
       CONTROL INPUT MEDIUM (ICONI), ONE PRINTER MEDIUM (IOUT), AND     NPM-0310
C     ONE RANDOM ACCESS FILE.  THIS RANDOM ACCESS FILE IS SIMULATED BY  NPM-0320
C     THE MODEL THROUGH CALLS TO DATA ACCESSING ROUTINE *DATA* (IE,     NPM-0330
C     *DATA* MIGHT MANEUVER A NUMBER OF SEQUENTIAL ACCESS FILES TO MEET NPM-0340
C     THE RANDOM REQUIREMENTS.  ALTERNATIVELY, *DATA* MIGHT MERELY      NPM-0350
C     TRANSFER DATA FROM A RANDOM ACESS FILE TO APPROPRIATE MEMORY      NPM-0360
C     LOCATIONS).  PROPER LOADING OF THIS RANDOM FILE (EITHER ACTUAL    NPM-0370
C     OR SIMULATED) IS THE USER=S RESPONSIBILITY.                       NPM-0380
C                                                                       NPM-0390
C     ALL RUN-TIME OUTPUTS ARE CONTROLLED BY PRINTER SWITCHES RESIDING  NPM-0400
C     IN LOGICAL ARRAY *PR*.  SWITCHES MAY BE RESET PERIODICALLY BY     NPM-0410
C     CONTROL COMMAND DURING THE RUN.  ALL SWITCHES ARE INITIALIZED OFF.NPM-0420
C                                                                       NPM-0430
C     PRINTER CONTROJS                                                  NPM-0440
C                                                                       NPM-0450
C     1 - PRINTS ALL INPUT DATA MATRICES BEFORE CALLING MAYNE           NPM-0460
C     2 - PRINTS U AND X MATRICES AFTER RETURNING FROM MAYNE            NPM-0470
C     3 - PRINTS STATISTICS FROM CRITER                                 NPM-0480
C     4 - PRINTS CRITERION FUNCTION VALUES FOR LPP TIME PERIODS         NPM-0490
C     5 - PRINTS JJ VALUE (SUM OF CRITERION FUNCTION VALUES)            NPM-0500
C     6 - PRINTS ESTIMATED DECREASE IN J-VALUES                         NPM-0510
C     7 - PRINTS CALCULATED X AND U MATRICES FOR EACH ITERATION IN MAYNENPM-0520
```

BEST COPY AVAILABLE


```
C     8 - PRINTS ADDITIONS TO TRANSFER MATRIX                          NPM-0530
C     9 - PRINTS AVERAGE ABILITY FOR EACH SECTOR                       NPM-0540
C     10 - PRINTS AVERAGE COST/STUDENT FOR EACH SECTOR                 NPM-0550
C     11 - PRINTS NUMBER OF GRADUATING HIGH SCHOOL STUDENTS            NPM-0560
C     12 - PRINTS SUPPLY MATRIX                                        NPM-0570
C     14 - PRINTS  P (PROBABILITY MATRIX)                              NPM-0580
C     13 - PRINTS NHS DECOMPOSITION MATRIX                             NPM-0590
C     15 - PRINTS D (DEMAND MATRIX)                                    NPM-0600
C     16 - PRINTS ENROLLMENT MATRIX                                    NPM-0610
C     17 - PRINTS NEW X,U, AND V VALUES AFTER E IS DETERMINED          NPM-0620
C     18 - PRINTS INPUT TRANSFER VECTOR                                NPM-0630
C     19 - PRINTS TRACER BENCHMARKS                                    NPM-0640
C     20 - ENABLES BINARY DUMP                                         NPM-0650
C     21 - INDICATES INTERACTIVE RUN                                   NPM-0660
C     22 - PRINTS TRANSITION MATRICES IN EQUATION FORMAT               NPM-0670
C     23 - PRINTS SECTOR INDEPENDENT,TIME DEPENDENT VARIABLES          NPM-0680
C     24 - PRINTS ALPHA,BETA                                           NPM-0690
C     25 - PRINTS VXX,VUX,VUU                                          NPM-0700
C     26 - PRINTS A,B,C                                                NPM-0710
C     27 - PRINTS VXX2,VX2                                             NPM-0720
C     28 - PRINTS CINV                                                 NPM-0730
C     29 - PRINTS HU,VU,VX                                             NPM-0740
C     30 - PRINTS CRITERION FUNCTION VALUES BY TERM                    NPM-0750
C                                                                      NPM-0760
C                                                                      NPM-0770
      REAL NHLAST,NHS                                                  NPM-0780
      LOGICAL TRANSF                                                   NPM-0790
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),         NPM-0800
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                        NPM-0810
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),    NPM-0820
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG   NPM-0830
      COMMON Q(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                 NPM-0840
      COMMON TRAN(12,16)                                               NPM-0850
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                        NPM-0860
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                NPM-0870
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                          NPM-0880
      LOGICAL PR,TEST                                                  NPM-0890
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,           NPM-0900
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                          NPM-0910
      COMMON/NAMES/NUM(16)                                             NPM-0920
      COMMON/VUDAT/ENROL(6,17),DIF(6,17),FTSTU(6,17),STATS(6,6)        NPM-0930
      COMMON/DEMSUP/D(16,13),S(16,13)                                  NPM-0940
      COMMON/SAVE/SAVE(16,6)                                           NPM-0950
C                                                                      NPM-0960
      DIMENSION P(16,13), E(16,13)                                     NPM-0970
      DIMENSION XFERS(12,5,16)                                         NPM-0980
      DIMENSION FA(12)                                                 NPM-0990
      REAL ISUM                                                        NPM-1000
C                                                                      NPM-1010
C     FOR AN INTERACTIVE JOB THE CONTROL OUTPUT AND DATA OUTPUT UNITS  NPM-1020
C     ARE SEPARATED (USUALLY THESE UNITS WILL CORRESPOND TO TELETYPE   NPM-1030
C     AND PRINTER RESPECTIVELY).  FOR A BATCH JOB THE TWO UNITS ARE    NPM-1040
```

BEST COPY AVAILABLE


```
C     EQUIVALENCED TO CHANNEL ICONO.                                    NPM-1050
C                                                                       NPM-1060
C                                                                       NPM-1070
C     INITIALIZATION                                                    NPM-1080
C                                                                       NPM-1090
      IF (PR(19)) WRITE (ICONO,1190) NUM(1)                             NPM-1100
      LPP=NPP+1                                                         NPM-1110
      JSECT=NSECT+1                                                     NPM-1120
      CALL DATA (1)                                                     NPM-1130
C                                                                       NPM-1140
C     OUTERMOST LOOP ON A FEDERAL PLANNING TARGET YEAR.                 NPM-1150
C                                                                       NPM-1160
      DO 720 IYR=1,3                                                    NPM-1170
        IARG=0                                                          NPM-1180
        CALL CONTRL (IARG)                                              NPM-1190
C     CODETO ZERO ARRAYS IN EVENT OF PARTIAL RUN OF LESS THAN NSECT     NPM-1200
C     SECTORS                                                           NPM-1210
        DO 110 J=1,NSECT                                                NPM-1220
          ABAR(J)=0.0                                                   NPM-1230
          DO 110 I=1,16                                                 NPM-1240
          CHAT(I,J)=0.0                                                 NPM-1250
          S(I,J)=0.0                                                    NPM-1260
          D(I,J)=0.0                                                    NPM-1270
          E(I,J)=0.0                                                    NPM-1280
          P(I,J)=0.0                                                    NPM-1290
110       CONTINUE                                                      NPM-1300
C     ZERO TRANSFER VECTOR                                              NPM-1310
        DO 120 I=1,5                                                    NPM-1320
          DO 120 J=1,12                                                 NPM-1330
          DO 120 K=1,16                                                 NPM-1340
120       XFERS(J,I,K)=0.0                                              NPM-1350
C     DUMP GLOBAL TIME-DEPENDENT VARIABLES                              NPM-1360
        IF (.NOT.PR(23)) GO TO 130                                      NPM-1370
        WRITE (IOUT,890)                                                NPM-1380
        WRITE (IOUT,940) NHLAST,(NHS(I),I=1,7)                          NPM-1390
        WRITE (IOUT,900) PCT                                            NPM-1400
        WRITE (IOUT,910) ACH                                            NPM-1410
        WRITE (IOUT,920) BETA1,BETA2                                    NPM-1420
        WRITE (IOUT,930) Y                                              NPM-1430
130     CONTINUE                                                        NPM-1440
C                                                                       NPM-1450
C                                                                       NPM-1460
C     MAIN LOOP ON NUMBER OF SECTORS--THIS LOOP COMPRISES THE           NPM-1470
C     INSTITUTIONAL SECTOR OF THE MODEL                                 NPM-1480
C                                                                       NPM-1490
C                                                                       NPM-1500
        DO 340 ISECT=1,NSECT                                            NPM-1510
          CALL CONTRL (IARG)                                            NPM-1520
          IF (IARG.GT.0) GO TO 350                                      NPM-1530
          IF (PR(19)) WRITE (ICONO,1010) IYR,ISECT                      NPM-1540
          CALL DATA (2)                                                 NPM-1550
C                                                                       NPM-1560
```

BEST COPY AVAILABLE


```
C     KEEP TRACK OF FRESHMEN ATTENDING                                  NPM-1570
C                                                                       NPM-1580
      IF (PR(19)) WRITE (ICONO,1190) NUM(2)                             NPM-1590
      TEMP=0.0                                                          NPM-1600
      DO 140 I=7,22                                                     NPM-1610
140     TEMP=TEMP+U(I,1)                                                NPM-1620
      FA(ISECT)=TEMP                                                    NPM-1630
C                                                                       NPM-1640
C     MOVE IN PENDING TRANSFERS                                         NPM-1650
C                                                                       NPM-1660
      IF (PR(19)) WRITE (ICONO,1190) NUM(3)                             NPM-1670
      IF (.NOT.PR(18)) GO TO 160                                        NPM-1680
      WRITE (IOUT,1170)                                                 NPM-1690
      WRITE (IOUT,770) (I,I=1,NPP)                                      NPM-1700
      DO 150 I=1,16                                                     NPM-1710
        WRITE (IOUT,1080) I,(XFERS(ISECT,J,I),J=1,NPP)                  NPM-1720
150     CONTINUE                                                        NPM-1730
160   CONTINUE                                                          NPM-1740
      DO 170 I=1,16                                                     NPM-1750
        DO 170 J=1,NPP                                                  NPM-1760
170     Z(I,J)=Z(I,J)+XFERS(ISECT,J,I)                                  NPM-1770
C                                                                       NPM-1780
C     COMPLETE X AND U MATRICES                                         NPM-1790
C                                                                       NPM-1800
      IF (PR(19)) WRITE (ICONO,1190) NUM(4)                             NPM-1810
      DO 180 J=1,NPP                                                    NPM-1820
        DO 180 I=1,NU                                                   NPM-1830
180     U(I,J)=U(I,1)                                                   NPM-1840
      DO 190 L=1,NPP                                                    NPM-1850
        M=L+1                                                           NPM-1860
        CALL NEWX (X(1,M),X(1,L),U(1,L),Z(1,L),L)                       NPM-1870
190     CONTINUE                                                        NPM-1880
C                                                                       NPM-1890
C                                                                       NPM-1900
      IF (.NOT.PR(1)) GO TO 230                                         NPM-1910
      WRITE (IOUT,730) ISECT                                            NPM-1920
      WRITE (IOUT,840)                                                  NPM-1930
      WRITE (IOUT,810) (INDIF(I),INDJF(I),VALF(I),I=1,NUMF)             NPM-1940
      WRITE (IOUT,850)                                                  NPM-1950
      WRITE (IOUT,820) (INDIG(I),INDJG(I),VALG(I),I=1,NUMG)             NPM-1960
      WRITE (IOUT,860)                                                  NPM-1970
      WRITE (IOUT,830) (H(I),I=1,NH)                                    NPM-1980
      WRITE (IOUT,760)                                                  NPM-1990
      WRITE (IOUT,770) (I,I=1,NPP)                                      NPM-2000
      DO 200 I=1,NZ                                                     NPM-2010
200     WRITE (IOUT,780) I,(Z(I,J),J=1,NPP)                             NPM-2020
      CALL UXDUMP                                                       NPM-2030
      WRITE (IOUT,790)                                                  NPM-2040
      WRITE (IOUT,770) (I,I=1,NPP)                                      NPM-2050
      DO 210 I=1,NQ                                                     NPM-2060
210     WRITE (IOUT,870) I,(Q(I,J),J=1,NPP)                             NPM-2070
      WRITE (IOUT,800)                                                  NPM-2080
```

BEST COPY AVAILABLE


```
      WRITE (IOUT,770) (I,I=1,NPP)                                      NPM-2090
      DO 220 I=1,NV                                                     NPM-2100
220     WRITE (IOUT,880) I,(VMS(I,J),J=1,NPP)                           NPM-2110
      WRITE (IOUT,960) (R(I),I=1,NTERMS)                                NPM-2120
      WRITE (IOUT,950) (WT(I),I=1,NTERMS)                               NPM-2130
      WRITE (IOUT,970) SPACE                                            NPM-2140
230   CONTINUE                                                          NPM-2150
C                                                                       NPM-2160
C                                                                       NPM-2170
      IF (.NOT.TEST) CALL MAYNE                                         NPM-2180
C                                                                       NPM-2190
      IF (PR(2)) CALL UXDUMP                                            NPM-2200
C                                                                       NPM-2210
C     CALCULATE AVERAGE ABILITY FOR CURRENT SECTOR IN FIRST TIME PERIOD  NPM-2220
C     NOTE--GRAD STUDENTS ARE WHOLLY OMITTED FROM THIS CALCULATION      NPM-2230
C                                                                       NPM-2240
      IF (PR(19)) WRITE (ICONO,1190) NUM(5)                             NPM-2250
      SUM=0.0                                                           NPM-2260
      TEMP=0.0                                                          NPM-2270
      M=13                                                              NPM-2280
      N=7                                                               NPM-2290
      DO 240 I=1,16                                                     NPM-2300
        XS=X(M,1)+X(M+16,1)+U(N,1)+Z(I,1)                               NPM-2310
        SUM=SUM+XS                                                      NPM-2320
        N=N+1                                                           NPM-2330
        M=M+1                                                           NPM-2340
240     TEMP=TEMP+ACH(I)*XS                                             NPM-2350
      ABAR(ISECT)=TEMP/SUM                                              NPM-2360
C                                                                       NPM-2370
C     CALCULATE APPARENT COST-TO-STUDENT VECTOR FOR THIS SECTOR.  ONLY  NPM-2380
C     COSTS TO INCOMING  FRESHMEN ARE OF INTEREST HERE                  NPM-2390
C                                                                       NPM-2400
      IF (PR(19)) WRITE (ICONO,1190) NUM(6)                             NPM-2410
      M=0                                                               NPM-2420
      DO 250 I=27,30                                                    NPM-2430
        TEMP1=U(I,1)+U(35,1)                                            NPM-2440
        DO 250 J=31,34                                                  NPM-2450
        M=M+1                                                           NPM-2460
250     VX(M)=TEMP1+U(J,1)                                              NPM-2470
      TEMP1=VMS(67,1)                                                   NPM-2480
      TEMP2=U(38,1)                                                     NPM-2490
      DO 260 I=1,16                                                     NPM-2500
260     CHAT(I,ISECT)=TEMP2-TEMP1*VMS(I,1)-VMS(I+33,1)+VX(I)            NPM-2510
C                                                                       NPM-2520
C     CALCULATE TRANSFER VECTOR                                         NPM-2530
C                                                                       NPM-2540
      IF (PR(19)) WRITE (ICONO,1190) NUM(7)                             NPM-2550
      IF (.NOT.TRANSF) GO TO 320                                        NPM-2560
      CALL DATA (9)                                                     NPM-2570
      DO 290 IT=1,NPP                                                   NPM-2580
        M=1                                                             NPM-2590
        DO 280 I=1,16                                                   NPM-2600
```

BEST COPY AVAILABLE


```
                DO 270 ISS=1,NSECT                                      NPM-2610
270               XFERS(ISS,IT,M)=TRAN(ISS,M)*SAVE(I,IT)+XFERS(ISS,IT,M)  NPM-2620
280             M=M+1                                                   NPM-2630
290           CONTINUE                                                  NPM-2640
C       PRINT UPDATED TRANSFER MATRIX IF REQUESTED                      NPM-2650
          IF (.NOT.PR(8)) GO TO 320                                     NPM-2660
          DO 310 IT=1,NPP                                               NPM-2670
            WRITE (IOUT,1160) IT                                        NPM-2680
            WRITE (IOUT,740) (J,J=1,NSECT)                              NPM-2690
            DO 300 I=1,16                                               NPM-2700
300           WRITE (IOUT,750) I,(XFERS(J,IT,I),J=1,NSECT)              NPM-2710
310         CONTINUE                                                    NPM-2720
320       CONTINUE                                                      NPM-2730
C                                                                       NPM-2740
C       COLLECT DATA FOR SUPPLY MATRIX S.  THIS MERELY CONSISTS OF      NPM-2750
C       GRABBING SELECTED CONTROL VARIABLES FOR TIME PERIOD 1           NPM-2760
C                                                                       NPM-2770
          IF (PR(19)) WRITE (ICONO,1190) NUM(8)                         NPM-2780
          M=7                                                           NPM-2790
          DO 330 I=1,16                                                 NPM-2800
            S(I,ISECT)=U(M,1)                                           NPM-2810
330         M=M+1                                                       NPM-2820
C                                                                       NPM-2830
          CALL DATA (3)                                                 NPM-2840
C                                                                       NPM-2850
          IF (PR(19)) WRITE (ICONO,1180) ISECT                          NPM-2860
340       CONTINUE                                                      NPM-2870
        IARG=NSECT                                                      NPM-2880
C                                                                       NPM-2890
C                                                                       NPM-2900
C       BEGIN STUDENT SECTOR OF MODEL                                   NPM-2910
C                                                                       NPM-2920
          CALL CONTRL (IARG)                                            NPM-2930
C                                                                       NPM-2940
C                                                                       NPM-2950
C       COMPLETE AV ABILITY / SECTOR VECTOR AND PRINT                   NPM-2960
C                                                                       NPM-2970
350       CONTINUE                                                      NPM-2980
          IF (PR(19)) WRITE (ICONO,1190) NUM(9)                         NPM-2990
          ISUM=0                                                        NPM-3000
          SUM=0.0                                                       NPM-3010
          DO 360 J=1,NSECT                                              NPM-3020
            ISUM=ISUM+FA(J)                                             NPM-3030
360         SUM=SUM+FA(J)*ABAR(J)                                       NPM-3040
          ABAR(JSECT)=(500.0*NHLAST-SUM)/(NHLAST-ISUM)                  NPM-3050
          IF (.NOT.PR(9)) GO TO 370                                     NPM-3060
          WRITE (IOUT,1110)                                             NPM-3070
          WRITE (IOUT,1050) (J,J=1,JSECT)                               NPM-3080
          WRITE (IOUT,1120) (ABAR(J),J=1,JSECT)                         NPM-3090
C                                                                       NPM-3100
C       COMPLETE COST-TO -STUDENT MATRIX AND PRINT                      NPM-3110
C                                                                       NPM-3120
```

BEST COPY AVAILABLE


```
370     CONTINUE                                                        NPM-3130
        IF (PR(19)) WRITE (ICONO,1190) NUM(10)                          NPM-3140
        DO 380 I=1,16                                                   NPM-3150
380       CHAT(I,JSECT)=0.0                                             NPM-3160
        IF (.NOT.PR(10)) GO TO 400                                      NPM-3170
        WRITE (IOUT,1130)                                               NPM-3180
        WRITE (IOUT,1050) (J,J=1,JSECT)                                 NPM-3190
        DO 390 I=1,16                                                   NPM-3200
          WRITE (IOUT,1080) I,(CHAT(I,J),J=1,JSECT)                     NPM-3210
390       CONTINUE                                                      NPM-3220
C                                                                       NPM-3230
C     PERFORM DECOMPOSITION ON NUMBER OF HIGH SCHOOL GRADUATES          NPM-3240
C                                                                       NPM-3250
400     CONTINUE                                                        NPM-3260
        IF (PR(19)) WRITE (ICONO,1190) NUM(11)                          NPM-3270
        IF (PR(11)) WRITE (IOUT,1020) NHS(1)                            NPM-3280
        DO 410 I=1,16                                                   NPM-3290
410       HS(I)=PCT(I)*NHS(IYR)                                         NPM-3300
        IF (PR(13)) WRITE (IOUT,1030) (I,HS(I),I=1,16)                  NPM-3310
C                                                                       NPM-3320
C     CALCULATE AND PRINT P MATRIX                                      NPM-3330
C                                                                       NPM-3340
        IF (PR(19)) WRITE (ICONO,1190) NUM(12)                          NPM-3350
        M=0                                                             NPM-3360
        DO 440 I=1,4                                                    NPM-3370
          DO 440 K=1,4                                                  NPM-3380
          M=M+1                                                         NPM-3390
          SUM=0.0                                                       NPM-3400
          ALPH=BETA2(I)*ACH(M)/1000.0                                   NPM-3410
          DO 420 J=1,JSECT                                              NPM-3420
            P(M,J)=EXP(ALPH*ABAR(J)+BETA1(I)*CHAT(M,J)/Y(M))            NPM-3430
420         SUM=SUM+P(M,J)                                              NPM-3440
          DO 430 J=1,JSECT                                              NPM-3450
430         P(M,J)=P(M,J)/SUM                                           NPM-3460
440       CONTINUE                                                      NPM-3470
        IF (.NOT.PR(14)) GO TO 460                                      NPM-3480
        WRITE (IOUT,1140)                                               NPM-3490
        WRITE (IOUT,1050) (J,J=1,JSECT)                                 NPM-3500
        DO 450 I=1,16                                                   NPM-3510
          WRITE (IOUT,1040) I,(P(I,J),J=1,JSECT)                        NPM-3520
450       CONTINUE                                                      NPM-3530
C                                                                       NPM-3540
C     CALCULATE DEMAND MATRIX                                           NPM-3550
C                                                                       NPM-3560
460     CONTINUE                                                        NPM-3570
        IF (PR(19)) WRITE (ICONO,1190) NUM(13)                          NPM-3580
        DO 470 I=1,16                                                   NPM-3590
          TEMP=HS(I)                                                    NPM-3600
          DO 470 J=1,JSECT                                              NPM-3610
470       D(I,J)=P(I,J)*TEMP                                            NPM-3620
        IF (.NOT.PR(15)) GO TO 490                                      NPM-3630
        WRITE (IOUT,1100)                                               NPM-3640
```

BEST COPY AVAILABLE


```
      WRITE (IOUT,1050) (J,J=1,JSECT)                                   NPM-3650
      DO 480 I=1,16                                                     NPM-3660
        WRITE (IOUT,1080) I,(D(I,J),J=1,JSECT)                          NPM-3670
480     CONTINUE                                                        NPM-3680
C                                                                       NPM-3690
C     PRINT SUPPLY MATRIX IF REQUESTED                                  NPM-3700
C                                                                       NPM-3710
490     CONTINUE                                                        NPM-3720
        IF (PR(19)) WRITE (ICONO,1190) NUM(14)                          NPM-3730
        IF (.NOT.PR(12)) GO TO 510                                      NPM-3740
        WRITE (IOUT,1090)                                               NPM-3750
        WRITE (IOUT,1050) (J,J=1,NSECT)                                 NPM-3760
        DO 500 I=1,16                                                   NPM-3770
          WRITE (IOUT,1080) I,(S(I,J),J=1,NSECT)                        NPM-3780
500       CONTINUE                                                      NPM-3790
C                                                                       NPM-3800
C     CALCULATE ENROLLMENT MATRIX                                       NPM-3810
C                                                                       NPM-3820
510     CONTINUE                                                        NPM-3830
        IF (PR(19)) WRITE (ICONO,1190) NUM(15)                          NPM-3840
        DO 520 J=1,NSECT                                                NPM-3850
          DO 520 I=1,16                                                 NPM-3860
520       E(I,J)=AMIN1(D(I,J),S(I,J))                                   NPM-3870
        DO 530 I=1,16                                                   NPM-3880
530       E(I,JSECT)=D(I,JSECT)                                         NPM-3890
        IF (.NOT.PR(16)) GO TO 550                                      NPM-3900
        WRITE (IOUT,1070)                                               NPM-3910
        WRITE (IOUT,1050) (J,J=1,JSECT)                                 NPM-3920
        DO 540 I=1,16                                                   NPM-3930
          WRITE (IOUT,1080) I,(E(I,J),J=1,JSECT)                        NPM-3940
540       CONTINUE                                                      NPM-3950
C                                                                       NPM-3960
C                                                                       NPM-3970
C     THE FOLLOWING SECTION OF THE MODEL INTRODUCES THE CALCULATED      NPM-3980
C     ENROLLMENT VALUES INTO THE CONTROL ARRAY, THUS VOIDING THE        NPM-3990
C     INSTITUTIONAL SECTOR CALCULATIONS.  THIS UPDATE IS DONE SECTOR    NPM-4000
C     BY SECTOR, WITH ALL I/O BEING HANDLED BY *DATA*.                  NPM-4010
C                                                                       NPM-4020
550     CONTINUE                                                        NPM-4030
        IF (PR(19)) WRITE (ICONO,980)                                   NPM-4040
        CALL DATA (6)                                                   NPM-4050
C     REZERO THE TRANSFER MATRIX                                        NPM-4060
        DO 560 I=1,5                                                    NPM-4070
          DO 560 J=1,12                                                 NPM-4080
          DO 560 K=1,16                                                 NPM-4090
560       XFERS(J,I,K)=0.0                                              NPM-4100
C     LOOP ON THE NUMBER OF ACTIVE SECTORS.                             NPM-4110
        DO 710 ISECT=1,IARG                                             NPM-4120
          IF (PR(19)) WRITE (ICONO,990) ISECT                           NPM-4130
          CALL DATA (7)                                                 NPM-4140
C     OVERRIDE OLD U-VALUES WITH ENROLLMENT VALUES.                     NPM-4150
          M=7                                                           NPM-4160
```

BEST COPY AVAILABLE


```
          DO 570 I=1,16                                                    NPM-4170
            U(M,1)=E(I,ISECT)                                              NPM-4180
570         M=M+1                                                          NPM-4190
          IF (.NOT.PR(18)) GO TO 590                                       NPM-4200
          WRITE (IOUT,1170)                                                NPM-4210
          WRITE (IOUT,770) (I,I=1,NPP)                                     NPM-4220
          DO 580 I=1,16                                                    NPM-4230
            WRITE (IOUT,1080) I,(XFERS(ISECT,J,I),J=1,NPP)                 NPM-4240
580         CONTINUE                                                       NPM-4250
590       CONTINUE                                                         NPM-4260
C     ADD THE TRANSFERS INTO THE EXOGENEOUS VARIABLES FOR EACH TIME        NPM-4270
C     PERIOD                                                               NPM-4280
          DO 600 I=1,16                                                    NPM-4290
            DO 600 J=1,NPP                                                 NPM-4300
600         Z(I,J)=Z(I,J)+XFERS(ISECT,J,I)                                 NPM-4310
C     RECALCULATE STATE VARIABLES FOR EACH TIME PERIOD USING NEW           NPM-4320
C     CONTROL VALUES                                                       NPM-4330
          DO 610 L=1,NPP                                                   NPM-4340
            M=L+1                                                          NPM-4350
            CALL NEWX (X(1,M),X(1,L),U(1,L),Z(1,L),L)                      NPM-4360
610         CONTINUE                                                       NPM-4370
          IF (.NOT.PR(5)) GO TO 630                                        NPM-4380
C     EXAMINE EFFECT ON THE LOSS FUNCTION FOR EACH TIME PERIOD.            NPM-4390
          SUM=0.0                                                          NPM-4400
          DO 620 I=1,LPP                                                   NPM-4410
            CALL CRITER (I)                                                NPM-4420
620         SUM=SUM+V(I)                                                   NPM-4430
          WRITE (IOUT,1060) ISECT,SUM                                      NPM-4440
630       CONTINUE                                                         NPM-4450
C     IF THE CURRENT SECTOR PRODUCES TRANSFERS, PERFORM THE MAPPING.       NPM-4460
          IF (.NOT.TRANSF) GO TO 700                                       NPM-4470
          CALL DATA (9)                                                    NPM-4480
          DO 660 IT=1,NPP                                                  NPM-4490
            M=1                                                            NPM-4500
            DO 650 I=1,16                                                  NPM-4510
              DO 640 ISS=1,NSECT                                           NPM-4520
640             XFERS(ISS,IT,M)=TRAN(ISS,M)*SAVE(I,IT)+XFERS(ISS,IT,M)     NPM-4530
650           M=M+1                                                        NPM-4540
660         CONTINUE                                                       NPM-4550
          IF (.NOT.PR(8)) GO TO 690                                        NPM-4560
          DO 680 IT=1,NPP                                                  NPM-4570
            WRITE (IOUT,1160) IT                                           NPM-4580
            WRITE (IOUT,740) (J,J=1,NSECT)                                 NPM-4590
            DO 670 I=1,16                                                  NPM-4600
670           WRITE (IOUT,750) I,(XFERS(J,IT,I),J=1,NSECT)                 NPM-4610
680         CONTINUE                                                       NPM-4620
690       CONTINUE                                                         NPM-4630
700       CONTINUE                                                         NPM-4640
          IF (PR(17)) CALL UXDUMP                                          NPM-4650
C     THE FOLLOWING CALL TO *DATA* WILL MOVE ALL DATA UP ONE               NPM-4660
C     TIME PERIOD AND RESTORE THESE VALUES FOR USE IN THE NEXT FEDERAL     NPM-4670
C     PLANNING YEAR.                                                       NPM-4680
```

BEST COPY AVAILABLE


```
          IF (IYR.EQ.1) CALL VIEWGO                                          NPM-4690
          CALL DATA (8)                                                      NPM-4700
        IF (PR(19)) WRITE (ICONO,1000) ISECT                                 NPM-4710
710       CONTINUE                                                           NPM-4720
        CALL DATA (4)                                                        NPM-4730
720     CONTINUE                                                             NPM-4740
      WRITE (ICONO,1150)                                                     NPM-4750
      STOP                                                                   NPM-4760
C                                                                            NPM-4770
C                                                                            NPM-4780
C                                                                            NPM-4790
730   FORMAT (////////,23H DATA FOR SECTOR NUMBER,I3)                        NPM-4800
740   FORMAT (/(3X,9I13))                                                    NPM-4810
750   FORMAT (1X,I2,9E13.3,/(3X,9E13.3))                                     NPM-4820
760   FORMAT (////,10H Z-MATRIX ,/)                                          NPM-4830
770   FORMAT (/(3X,10I12))                                                   NPM-4840
780   FORMAT (2X,2HZ(,I2,2H) ,8G12.4)                                        NPM-4850
790   FORMAT (////,10H Q-MATRIX ,/)                                          NPM-4860
800   FORMAT (////,12H VMS-MATRIX ,/)                                        NPM-4870
810   FORMAT (4(6X,2HF(,I2,1H,,I2,4H) = ,G12.3))                             NPM-4880
820   FORMAT (4(6X,2HG(,I2,1H,,I2,4H) = ,G12.3))                             NPM-4890
830   FORMAT (10E12.1)                                                       NPM-4900
840   FORMAT (////,10H F-MATRIX ,/)                                          NPM-4910
850   FORMAT (////,10H G-MATRIX ,/)                                          NPM-4920
860   FORMAT (////,10H H-VECTOR ,/)                                          NPM-4930
870   FORMAT (2X,2HQ(,I2,2H) ,8G12.4)                                        NPM-4940
880   FORMAT (1X,4HVMS(,I2,2H) ,8G12.4)                                      NPM-4950
890   FORMAT (////,39H GLOBAL TIME DEPENDENT VARIABLES FOLLOW,//)            NPM-4960
900   FORMAT (////,44H HIGH SCHOOL DECOMPOSITION PERCENTAGES (A,I),16(/,NPM-4970
     15X,F10.2))                                                             NPM-4980
910   FORMAT (////,25H HS ACHIEVEMENT SAT (A,I),/(5X,F10.2))                 NPM-4990
920   FORMAT (////,18HBETA1 COEFFICIENTS,4E15.2/,18HBETA2 COEFFICIENTS,4NPM-5000
     1E15.2)                                                                 NPM-5010
930   FORMAT (////,13H INCOME (A,I),16(/,5X,F10.2))                          NPM-5020
940   FORMAT (////,40H HIGH SCHOOL STUDENTS FOR PERIOD 0...NPP,12(/,10X,NPM-5030
     1E20.5))                                                                NPM-5040
950   FORMAT (////,8H WEIGHTS,15(/,10G12.2))                                 NPM-5050
960   FORMAT (////,8H TARGETS,15(/,10G12.2))                                 NPM-5060
970   FORMAT (////,19H SPACE COEFFICIENTS,15(/,10G12.2))                     NPM-5070
980   FORMAT (59H1......RECALCULATE STATE VARIABLES BASED ON ENROLLMENT NPM-5080
     1INFO,//)                                                               NPM-5090
990   FORMAT (////,39H ......BEGIN RECALCULATIONS FOR SECTOR ,I3)            NPM-5100
1000  FORMAT (//,54H ......RECALCULATIONS AND UPDATES COMPLETED FOR SECTNPM-5110
     1OR,I3)                                                                 NPM-5120
1010  FORMAT(1H1,57H**********************************************PLANNING PERNPM-5130
     1IOD =,I5,* SECTOR = *,I5)                                              NPM-5140
1020  FORMAT (////,63H NUMBER OF GRADUATING HIGH SCHOOL STUDENTS FOR THINPM-5150
     1S TIME PERIOD,E15.5)                                                   NPM-5160
1030  FORMAT (////,43H HIGH SCHOOL STUDENTS BY INCOME AND ABILITY,16(/,INPM-5170
     16,5X,E15.5))                                                           NPM-5180
1040  FORMAT (/I4,9G12.3,20(/4X,9G12.3))                                     NPM-5190
1050  FORMAT (/(9I12))                                                       NPM-5200
```

BEST COPY AVAILABLE


```
1060  FORMAT (////,46H NEW TOTAL CRITERION FUNCTION VALUE FOR SECTOR,I3,NPM-5210
     14H IS ,E15.4)                                                    NPM-5220
1070  FORMAT (////,62H ENROLLMENT MATRIX - ROWS=STUDENT GROUPS, COLUMS=INPM-5230
     1NST. SECTORS,/)                                                  NPM-5240
1080  FORMAT (/I4,9G12.3,20(/,4X,9G12.3))                               NPM-5250
1090  FORMAT (////,56H SUPPLY MATRIX---ROWS=STUDENT GROUPS, COLS=INST. SNPM-5260
     1ECTORS,/)                                                        NPM-5270
1100  FORMAT (////,56H DEMAND MATRIX---ROWS=STUDENT GROUPS, COLS=INST. SNPM-5280
     1ECTORS,/)                                                        NPM-5290
1110  FORMAT (////,26H AVERAGE ABILITY BY SECTOR,/)                     NPM-5300
1120  FORMAT (/(4X,9G12.3))                                             NPM-5310
1130  FORMAT (////,73H APPARENT COST-TO-STUDENT MATRIX, ROWS=STUDENT GRONPM-5320
     1UPS, COLS=INST. SECTORS,/)                                       NPM-5330
1140  FORMAT (////,50H P-MATRIX, ROWS=STUDENT GROUPS, COLS=INST. SECTORSNPM-5340
     1,/)                                                              NPM-5350
1150  FORMAT (41H ......PLANNING MODEL TERMINATED NORMALLY)             NPM-5360
1160  FORMAT (////,27H TRANSFER MATRIX FOR PERIOD,I3,2X,40HROWS=STUDENT NPM-5370
     1GROUPS, COLS=INST.  SECTORS,/)                                   NPM-5380
1170  FORMAT (////,62H INPUT TRANSFER VECTOR, ROWS=STUDENT GROUPS, COLS=NPM-5390
     1TIME PERIODS,/)                                                  NPM-5400
1180  FORMAT (////,13H ......SECTOR,I3,11H COMPLETED.)                  NPM-5410
1190  FORMAT (17H ......NPM CKPT. ,I3)                                  NPM-5420
      END                                                               NPM-5430
```

BEST COPY AVAILABLE


```
      SUBROUTINE MAYNE                                                  MAY-0010
C                                                                       MAY-0020
C     THIS PROGRAM EMPLOYS AN ITERATIVE PROCEDURE TO FIND A SEQUENCE    MAY-0030
C     OF CONTROL VECTORS WHICH MINIMIZES A NON-LINEAR N-PERIOD OBJECTIVEMAY-0040
C     FUNCTION SUBJECT TO LINEAR-DYNAMIC CONSTRAINTS.  THE PROGRAM IS   MAY-0050
C     BASED ON AN ALGORITHM DEVELOPED BY DAVID MAYNE, PUBLISHED BY      MAY-0060
C     INT. JOURNAL OF CONTROL (1966).                                   MAY-0070
C                                                                       MAY-0080
      REAL NHLAST,NHS                                                   MAY-0090
      LOGICAL TRANSF                                                    MAY-0100
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),          MAY-0110
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                         MAY-0120
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),     MAY-0130
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG    MAY-0140
      COMMON C(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                  MAY-0150
      COMMON TRAN(12,16)                                                MAY-0160
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                         MAY-0170
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                 MAY-0180
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           MAY-0190
      LOGICAL PR,TEST                                                   MAY-0200
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            MAY-0210
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           MAY-0220
      COMMON/NAMES/NUM(16)                                              MAY-0230
      DIMENSION INEW(400)                                               MAY-0240
      DIMENSION ALPHA(38,5), BETA(38,49)                                MAY-0250
      DIMENSION A(49,49), B(38,49), C(38,38), CINV(38,38)               MAY-0260
      DIMENSION CC(1), XNEW(49,6), HU(38)                               MAY-0270
      DIMENSION T(49)                                                   MAY-0280
      DIMENSION WLIN(2401)                                              MAY-0290
      DIMENSION CLIN(1444)                                              MAY-0300
      EQUIVALENCE (WLIN,W)                                              MAY-0310
      EQUIVALENCE (T,XNEW)                                              MAY-0320
      EQUIVALENCE (BETA,CC,W)                                           MAY-0330
      EQUIVALENCE (A,VXX), (B,VUX), (C,VUU,CINV,CLIN)                   MAY-0340
      EQUIVALENCE (XNEW,INEW)                                           MAY-0350
C                                                                       MAY-0360
C     BEGIN ITERATIVE PROCEDURE                                         MAY-0370
C                                                                       MAY-0380
      CALL SECOND (YYYY)                                                MAY-0390
      IF (PR(19)) WRITE (ICONO,530) ISECT,YYYY                          MAY-0400
      NXNX=NX*NX                                                        MAY-0410
      NCYCLE=0                                                          MAY-0420
      CALL CALCV (0)                                                    MAY-0430
110   CONTINUE                                                          MAY-0440
      CALL DATA (5)                                                     MAY-0450
      CALL CALCV (LPP)                                                  MAY-0460
      SUM=0.0                                                           MAY-0470
      DO 120 I=1,LPP                                                    MAY-0480
        CALL CRITER (I)                                                 MAY-0490
120     SUM=SUM+V(I)                                                    MAY-0500
      IF (PR(5)) WRITE (IOUT,590) NCYCLE,SUM                            MAY-0510
C                                                                       MAY-0520
```

BEST COPY AVAILABLE


```
      IF (NCYCLE.LT.NUMIT) GO TO 130                                    MAY-0530
      CALL SECOND (YYYY)                                                MAY-0540
      IF (PR(19)) WRITE (ICUNO,560) YYYY,NCYCLE                         MAY-0550
      RETURN                                                            MAY-0560
C                                                                       MAY-0570
C     BEGIN MAJOR LOOP                                                  MAY-0580
C                                                                       MAY-0590
130   AA=0.0                                                            MAY-0600
      NCYCLE=NCYCLE+1                                                   MAY-0610
      IF (PR(19)) WRITE (ICONO,540) NCYCLE                              MAY-0620
      DO 460 II=1,NPP                                                   MAY-0630
        IT=LPP-II                                                       MAY-0640
        IF (PR(19)) WRITE (ICONO,550) IT                                MAY-0650
        CALL UPTRAN (IT)                                                MAY-0660
        IF (PR(22)) CALL DEBUG (1,IT,X(1,IT))                           MAY-0670
        CALL SECOND (TIME)                                              MAY-0680
        IF (PR(19)) WRITE (ICONO,600) NUM(1),TIME                       MAY-0690
        CALL CALCV (IT)                                                 MAY-0700
        CALL DATA (10)                                                  MAY-0710
        IF (PR(27)) CALL MDUMP (4HVXX2,VXX2,NX,NX,NX)                   MAY-0720
        IF (PR(27)) CALL VDUMP (3HVX2,VX2,NX)                           MAY-0730
        IF (PR(25)) CALL MDUMP (3HVXX,VXX,NX,NX,NX)                     MAY-0740
        IF (PR(25)) CALL MDUMP (3HVUX,VUX,NU,NU,NX)                     MAY-0750
        IF (PR(25)) CALL MDUMP (3HVUU,VUU,NU,NU,NU)                     MAY-0760
        IF (PR(29)) CALL VDUMP (2HVU,VU,NU)                             MAY-0770
        IF (PR(29)) CALL VDUMP (2HVX,VX,NX)                             MAY-0780
        DO 140 I=1,NXNX                                                 MAY-0790
140       WLIN(I)=0.0                                                   MAY-0800
        DO 160 I=1,NF                                                   MAY-0810
          IX=INDIF(I)                                                   MAY-0820
          IY=INDJF(I)                                                   MAY-0830
          DO 150 K=1,NX                                                 MAY-0840
150         W(K,IY)=W(K,IY)+VXX2(K,IX)*VALF(I)                          MAY-0850
160       CONTINUE                                                      MAY-0860
        CALL SECOND (TIME)                                              MAY-0870
        IF (PR(19)) WRITE (ICONO,600) NUM(2),TIME                       MAY-0880
C                                                                       MAY-0890
C     CALCULATE MATRIX A = VXX + F' VXX(T+1) F                          MAY-0900
C                                                                       MAY-0910
        DO 180 I=1,NF                                                   MAY-0920
          IX=INDIF(I)                                                   MAY-0930
          IY=INDJF(I)                                                   MAY-0940
          DO 170 K=1,NX                                                 MAY-0950
170         A(IY,K)=A(IY,K)+VALF(I)*W(IX,K)                             MAY-0960
180       CONTINUE                                                      MAY-0970
        IF (PR(26)) CALL MDUMP (1HA,A,NX,NX,NX)                         MAY-0980
        CALL SECOND (TIME)                                              MAY-0990
        IF (PR(19)) WRITE (ICONO,600) NUM(3),TIME                       MAY-1000
C                                                                       MAY-1010
C     CALCULATE MATRIX B = VUX + G' VXX(T+1) F                          MAY-1020
C                                                                       MAY-1030
        DO 200 I=1,NG                                                   MAY-1040
```

BEST COPY AVAILABLE


```
          IX=INDIG(I)                                                    MAY-1050
          IY=INDJG(I)                                                    MAY-1060
          DO 190 K=1,NX                                                  MAY-1070
190         B(IY,K)=B(IY,K)+VALG(I)*W(IX,K)                              MAY-1080
200       CONTINUE                                                       MAY-1090
        IF (PR(26)) CALL MDUMP (1HB,B,NU,NU,NX)                          MAY-1100
        CALL SECOND (TIME)                                               MAY-1110
        IF (PR(19)) WRITE (ICONO,600) NUM(4),TIME                        MAY-1120
C                                                                        MAY-1130
C     CALCULATE MATRIX C = VUU + G' VXX(T+1) G                           MAY-1140
C                                                                        MAY-1150
        DO 210 I=1,NXNX                                                  MAY-1160
210       WLIN(I)=0.0                                                    MAY-1170
        DO 230 I=1,NG                                                    MAY-1180
          IX=INDIG(I)                                                    MAY-1190
          IY=INDJG(I)                                                    MAY-1200
          DO 220 K=1,NX                                                  MAY-1210
220         W(K,IY)=W(K,IY)+VXX2(K,IX)*VALG(I)                           MAY-1220
230       CONTINUE                                                       MAY-1230
        DO 250 I=1,NG                                                    MAY-1240
          IX=INDIG(I)                                                    MAY-1250
          IY=INDJG(I)                                                    MAY-1260
          DO 240 K=1,NU                                                  MAY-1270
240         C(IY,K)=C(IY,K)+VALG(I)*W(IX,K)                              MAY-1280
250       CONTINUE                                                       MAY-1290
        IF (PR(26)) CALL MDUMP (1HC,C,NU,NU,NU)                          MAY-1300
        CALL SECOND (TIME)                                               MAY-1310
        IF (PR(19)) WRITE (ICONO,600) NUM(5),TIME                        MAY-1320
C                                                                        MAY-1330
C     PLACE UPPER TRIANGLE OF C INTO CC BY COLS.                         MAY-1340
C                                                                        MAY-1350
        K=1                                                              MAY-1360
        DO 280 J=1,NU                                                    MAY-1370
          DO 270 I=1,J                                                   MAY-1380
            IF (C(I,J).LT.0.0) GO TO 260                                 MAY-1390
            CC(K)=AMIN1(C(I,J),C(J,I))                                   MAY-1400
            GO TO 270                                                    MAY-1410
260         CC(K)=AMAX1(C(I,J),C(J,I))                                   MAY-1420
270         K=K+1                                                        MAY-1430
280       CONTINUE                                                       MAY-1440
        IER=0                                                            MAY-1450
        EPS=.000001                                                      MAY-1460
        CALL DSINV (CC,NU,EPS,IER)                                       MAY-1470
        CALL SECOND (TIME)                                               MAY-1480
        IF (PR(19)) WRITE (ICONO,600) NUM(6),TIME                        MAY-1490
        IF (IER.NE.-1) GO TO 290                                         MAY-1500
        WRITE (ICONO,580)                                                MAY-1510
        IF (LINEQU(C,INEW,NU,NU,76,1E-50).EQ.0) GO TO 320                MAY-1520
        WRITE (ICONO,570)                                                MAY-1530
        WRITE (IOUT,570)                                                 MAY-1540
        STOP                                                             MAY-1550
C                                                                        MAY-1560
```

BEST COPY AVAILABLE


```
C     PLACE CC IN CINV                                                  MAY-1570
C                                                                       MAY-1580
290     CONTINUE                                                        MAY-1590
        L=1                                                             MAY-1600
        DO 310 J=1,NU                                                   MAY-1610
          DO 300 I=1,J                                                  MAY-1620
            CINV(I,J)=CC(L)                                             MAY-1630
            CINV(J,I)=CC(L)                                             MAY-1640
300         L=L+1                                                       MAY-1650
310       CONTINUE                                                      MAY-1660
320     CONTINUE                                                        MAY-1670
        IF (PR(28)) CALL MDUMP (4HCINV,CINV,NU,NU,NU)                   MAY-1680
        DO 330 I=1,NU                                                   MAY-1690
330       HU(I)=VU(I)                                                   MAY-1700
        DO 340 I=1,NG                                                   MAY-1710
          IX=INDIG(I)                                                   MAY-1720
          IY=INDJG(I)                                                   MAY-1730
340       HU(IY)=HU(IY)+VALG(I)*VX2(IX)                                 MAY-1740
        IF (PR(29)) CALL VDUMP (2HHU,HU,NU)                             MAY-1750
        CALL SECOND (TIME)                                              MAY-1760
        IF (PR(19)) WRITE (ICONO,600) NUM(7),TIME                       MAY-1770
C                                                                       MAY-1780
C     CALCULATE ALPHA(IT),BETA(IT)                                      MAY-1790
C                                                                       MAY-1800
        DO 380 I=1,NU                                                   MAY-1810
          TEMP=0.0                                                      MAY-1820
          DO 350 J=1,NU                                                 MAY-1830
350         TEMP=TEMP+CINV(I,J)*HU(J)                                   MAY-1840
          ALPHA(I,IT)=-TEMP                                             MAY-1850
          DO 370 J=1,NX                                                 MAY-1860
            TEMP=0.0                                                    MAY-1870
            DO 360 K=1,NU                                               MAY-1880
360           TEMP=TEMP+CINV(I,K)*B(K,J)                                MAY-1890
370         BETA(I,J)=-TEMP                                             MAY-1900
380       CONTINUE                                                      MAY-1910
        CALL DATA (11)                                                  MAY-1920
        IF (PR(24)) CALL MDUMP (4HBETA,BETA,NU,NU,NX)                   MAY-1930
        IF (PR(24)) CALL VDUMP (4HALPH,ALPHA(1,IT),NU)                  MAY-1940
        CALL SECOND (TIME)                                              MAY-1950
        IF (PR(19)) WRITE (ICONO,600) NUM(8),TIME                       MAY-1960
C                                                                       MAY-1970
C     CALCULATE NEW VXX2 AND NEW VX2                                    MAY-1980
C                                                                       MAY-1990
        DO 390 I=1,NX                                                   MAY-2000
390       T(I)=0.0                                                      MAY-2010
        DO 400 I=1,NF                                                   MAY-2020
          IX=INDIF(I)                                                   MAY-2030
          IY=INDJF(I)                                                   MAY-2040
400       T(IY)=T(IY)+VALF(I)*VX2(IX)                                   MAY-2050
        DO 420 I=1,NX                                                   MAY-2060
          TEMP=0.0                                                      MAY-2070
          DO 410 J=1,NU                                                 MAY-2080
```

BEST COPY AVAILABLE


```
410           TEMP=TEMP+BETA(J,I)*HU(J)                                  MAY-2090
420         VX2(I)=T(I)+VX(I)+TEMP                                       MAY-2100
          DO 440 I=1,NX                                                  MAY-2110
            DO 440 J=1,NX                                                MAY-2120
            TEMP=0.0                                                     MAY-2130
            DO 430 K=1,NU                                                MAY-2140
430           TEMP=TEMP-BETA(K,J)*B(K,I)                                 MAY-2150
440         VXX2(I,J)=A(I,J)-TEMP                                        MAY-2160
          CALL SECOND (TIME)                                             MAY-2170
          IF (PR(19)) WRITE (ICONO,600) NUM(9),TIME                      MAY-2180
C       CALCULATE AA TO INDICATE EXPECTED IMPROVEMENT IN J-VALUE         MAY-2190
          TEMP=0.0                                                       MAY-2200
          DO 450 I=1,NU                                                  MAY-2210
450         TEMP=TEMP+HU(I)*ALPHA(I,IT)                                  MAY-2220
460       AA=AA+0.5*TEMP                                                 MAY-2230
C                                                                        MAY-2240
C       END LOOP CYCLING ON TIME PERIODS                                 MAY-2250
C                                                                        MAY-2260
        IF (PR(6)) WRITE (IOUT,520) AA                                   MAY-2270
C                                                                        MAY-2280
C       CALCULATE NEW CONTROLS AND NEW STATE VARIABLES                   MAY-2290
C                                                                        MAY-2300
        DO 470 I=1,NU                                                    MAY-2310
          U(I,1)=U(I,1)+ALPHA(I,1)                                       MAY-2320
470       CONTINUE                                                       MAY-2330
        CALL NEWX (XNEW(1,2),X(1,1),U(1,1),Z(1,1),1)                     MAY-2340
        DO 500 L=2,NPP                                                   MAY-2350
          CALL DATA (12)                                                 MAY-2360
          M=L+1                                                          MAY-2370
          DO 490 I=1,NU                                                  MAY-2380
            TEMP=0.0                                                     MAY-2390
            DO 480 J=1,NX                                                MAY-2400
480           TEMP=TEMP+BETA(I,J)*(XNEW(J,L)-X(J,L))                     MAY-2410
            TEMP=TEMP+ALPHA(I,L)+U(I,L)                                  MAY-2420
            U(I,L)=TEMP                                                  MAY-2430
490         CONTINUE                                                     MAY-2440
          CALL NEWX (XNEW(1,M),XNEW(1,L),U(1,L),Z(1,L),L)                MAY-2450
500       CONTINUE                                                       MAY-2460
        DO 510 I=1,NX                                                    MAY-2470
          DO 510 J=2,LPP                                                 MAY-2480
510       X(I,J)=XNEW(I,J)                                               MAY-2490
        IF (PR(7)) CALL UXDUMP                                           MAY-2500
        CALL SECOND (TIME)                                               MAY-2510
        IF (PR(19)) WRITE (ICONO,600) NUM(10),TIME                       MAY-2520
        GO TO 110                                                        MAY-2530
C                                                                        MAY-2540
C                                                                        MAY-2550
520     FORMAT (/,35H ESTIMATED DECREASE IN J-VALUES IS ,G15.4)          MAY-2560
530     FORMAT (29H ......ENTER MAYNE FOR SECTOR,I3,8H AT ET= ,F6.3)     MAY-2570
540     FORMAT (22H ......BEGIN ITERATION,I2)                            MAY-2580
550     FORMAT (////,1X,6(6H......),13H BEGIN CYCLE ,I3)                 MAY-2590
560     FORMAT (24H ......EXIT MAYNE AT ET=,F6.3,6H AFTER,I3,11H ITERATICNMAY-2600
```

BEST COPY AVAILABLE


```
     1S)                                                                MAY-2610
570   FORMAT (51H .....SUBROUTINE MAYNE ABORTED DUE TO BAD MATRIX C)    MAY-2620
580   FORMAT (60H .....C NOT POSITIVE DEFINITE. STANDARD INVERSION ATTEMAY-2630
     1MPTED)                                                            MAY-2640
590   FORMAT (6H AFTER,I3,46H ITERATIONS, TOTAL CRITERION FUNCTION VALUEMAY-2650
     1 IS,E15.4)                                                        MAY-2660
600   FORMAT (15H .....AT POINT,I3,4H ET=,F7.3)                          MAY-2670
      END                                                               MAY-2680
```

BEST COPY AVAILABLE


```
      SUBROUTINE NEWX (XNEW,XOLD,UOLD,ZOLD,K)                          NEW-0010
      DIMENSION XNEW(49), XOLD(49), UOLD(38), ZOLD(25)                 NEW-0020
C                                                                      NEW-0030
C     XNEW IS THE VECTOR OF VALUES RESULTING FROM NEWX                 NEW-0040
C     XOLD IS THE VECTOR OF OLD STATE VARIABLES                        NEW-0050
C     UOLD IS THE VECTOR OF OLD CONTROL VARIABLES                      NEW-0060
C     ZOLD IS VECTOR OF OLD EXOGENEOUS VARIABLES                       NEW-0070
C     K IS THE TIME PERIOD INVOLVED                                    NEW-0080
C                                                                      NEW-0090
C       N.B.---THE ABOVE VECTORS ARE ACTUALLY COLUMNS IN MATRICES.     NEW-0100
C     THEY HAVE BEEN PASSED AS VECTORS TO AVOID SENSELESS SUBSCRIPT    NEW-0110
C     CALCUALTIONS                                                     NEW-0120
C                                                                      NEW-0130
      LOGICAL PR,TEST                                                  NEW-0140
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,           NEW-0150
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                         NEW-0160
      REAL NHLAST,NHS                                                  NEW-0170
      LOGICAL TRANSF                                                   NEW-0180
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),         NEW-0190
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                        NEW-0200
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),    NEW-0210
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG   NEW-0220
      COMMON C(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                 NEW-0230
      COMMON TRAN(12,16)                                               NEW-0240
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                        NEW-0250
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                NEW-0260
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                          NEW-0270
      COMMON/NAMES/NUM(16)                                             NEW-0280
      COMMON/SAVE/SAVE(16,6)                                           NEW-0290
C                                                                      NEW-0300
C                                                                      NEW-0310
C     THIS ROUTINE PRODUCES A NEW SET OF STATE VARIABLES AS A FUNCTION NEW-0320
C     OF THE PRECEDING TIME PERIOD'S STATE AND CONTROL VALUES.  MANY   NEW-0330
C     OF THE CALCULATIONS HERE  ARE IDENTICAL TO THOSE IN *UPTRAN*.    NEW-0340
C     THAT ROUTINE, HOWEVER, SUPPLIES COEFFICIENTS FOR PARTIALS MATRICESNEW-0350
C  OF THE TRANSITION FUNCTION.  IN THIS ROUTINE, THE MATRIX FORMULATION NEW-0360
C     IS USED ONLY FOR CONSTANT COEFFICIENTS--OTHER ENTRIES ARE        NEW-0370
C     DISPLAYED AS LINE ITEMS FOR INCREASED COMPREHENSIBILITY.         NEW-0380
C                                                                      NEW-0390
C                                                                      NEW-0400
      IF (PR(22)) WRITE (IOUT,270) K                                   NEW-0410
      DO 110 I=1,NX                                                    NEW-0420
110     XNEW(I)=0.0                                                    NEW-0430
      DO 120 I=1,NUMF                                                  NEW-0440
        IX=INDIF(I)                                                    NEW-0450
        IY=INDJF(I)                                                    NEW-0460
120     XNEW(IX)=XNEW(IX)+VALF(I)*XOLD(IY)                             NEW-0470
      DO 130 I=1,NUMG                                                  NEW-0480
        IX=INDIG(I)                                                    NEW-0490
        IY=INDJG(I)                                                    NEW-0500
130     XNEW(IX)=XNEW(IX)+VALG(I)*UOLD(IY)                             NEW-0510
      DO 140 I=1,16                                                    NEW-0520
```

BEST COPY AVAILABLE


```
140     XNEW(I+28)=XNEW(I+28)+ZOLD(I)                                   NEW-0530
      XNEW(46)=XNEW(46)+ZOLD(17)                                        NEW-0540
      XNEW(47)=XNEW(47)+H(1)*ZOLD(18)                                   NEW-0550
      IF (PR(22)) CALL DEBUG (0,K,XNEW)                                 NEW-0560
C     COMMENCE MAKING ADDITIONS TO NCB CONSTRUCTION                     NEW-0570
      USUM=0.0                                                          NEW-0580
      XSUMHI=0.0                                                        NEW-0590
      XSUMLO=0.0                                                        NEW-0600
      ZSUM=0.0                                                          NEW-0610
      DO 150 I=1,16                                                     NEW-0620
        USUM=USUM+UOLD(I+6)                                             NEW-0630
        XSUMLO=XSUMLO+XOLD(I+12)                                        NEW-0640
150     ZSUM=ZSUM+ZOLD(I)                                               NEW-0650
      DO 160 I=29,44                                                    NEW-0660
160     XSUMHI=XSUMHI+XOLD(I)                                           NEW-0670
      TEMP1=H(2)*ZOLD(17)                                               NEW-0680
      TEMP2=Q(1,K)*(UOLD(5)+UOLD(6))                                    NEW-0690
      TEMP3=Q(2,K)*(USUM+XSUMLO)                                        NEW-0700
      TEMP4=Q(3,K)*(ZSUM+XSUMHI)                                        NEW-0710
      TEMP5=Q(4,K)*(XOLD(45)+UOLD(23))                                  NEW-0720
      TEMP6=Q(5,K)*(UOLD(K)+UOLD(6))                                    NEW-0730
      TEMP7=Q(6,K)*(USUM+XSUMLO)                                        NEW-0740
      TEMP8=Q(7,K)*(XSUMHI+SUM)                                         NEW-0750
      TEMP9=Q(8,K)*(XOLD(45)+UOLD(23))                                  NEW-0760
      TEMP=XNEW(48)                                                     NEW-0770
      XNEW(48)=XNEW(48)+TEMP1+TEMP2+TEMP3+TEMP4+TEMP5+TEMP6+TEMP7+TEMP8+NEW-0780
     1TEMP9+ZOLD(19)+ZOLD(20)                                           NEW-0790
      IF (PR(22)) WRITE (IOUT,240) TEMP,TEMP1,ZOLD(19),ZOLD(20),TEMP2,TENEW-0800
     1MP3,TEMP4,TEMP5,TEMP6,TEMP7,TEMP8,TEMP9,XNEW(48)                  NEW-0810
C                                                                       NEW-0820
C     PROCEED WITH NCB OPERATING                                        NEW-0830
      ASUM=USUM+ZSUM+XSUMHI+XSUMLO+UOLD(23)+XOLD(45)                    NEW-0840
      TEMP1=Q(10,K)*ASUM                                                NEW-0850
      TEMP2=Q(9,K)*ASUM                                                 NEW-0860
      TEMP3=UOLD(38)*ASUM                                               NEW-0870
      TEMP4=Q(43,K)*(UOLD(23)+XOLD(45))                                 NEW-0880
      DO 170 I=11,26                                                    NEW-0890
170     TEMP4=TEMP4+Q(I,K)*(XOLD(I+2)+UOLD(I-4))                        NEW-0900
      DO 180 I=27,43                                                    NEW-0910
180     TEMP4=TEMP4+Q(I,K)*(XOLD(I+2)+ZOLD(I-26))                       NEW-0920
      IF (PR(22)) WRITE (IOUT,250) XNEW(49),TEMP1,TEMP2,TEMP3,TEMP4,ZOLDNEW-0930
     1(21),ZOLD(22)                                                     NEW-0940
      ALL=0.0                                                           NEW-0950
      M=12                                                              NEW-0960
      N=6                                                               NEW-0970
      DO 200 I=27,30                                                    NEW-0980
        TEMP8=UOLD(35)+UOLD(I)                                          NEW-0990
        DO 190 J=31,34                                                  NEW-1000
          M=M+1                                                         NEW-1010
          N=N+1                                                         NEW-1020
          SUM=TEMP8+UOLD(J)                                             NEW-1030
          TOTAL=SUM*(XOLD(M)+UOLD(N))                                   NEW-1040
```

BEST COPY AVAILABLE


```
          ALL=ALL+TOTAL                                                    NEW-1050
          IF (PR(22)) WRITE (IOUT,260) TOTAL,SUM,XOLD(M),UOLD(N)           NEW-1060
190       CONTINUE                                                         NEW-1070
200     CONTINUE                                                           NEW-1080
        M=28                                                               NEW-1090
        N=0                                                                NEW-1100
        DO 220 I=27,30                                                     NEW-1110
          TEMP8=UOLD(36)+UOLD(I)                                           NEW-1120
          DO 210 J=31,34                                                   NEW-1130
            M=M+1                                                          NEW-1140
            N=N+1                                                          NEW-1150
            SUM=TEMP8+UOLD(J)                                              NEW-1160
            TOTAL=SUM*(XOLD(M)+ZOLD(N))                                    NEW-1170
            ALL=ALL+TOTAL                                                  NEW-1180
            IF (PR(22)) WRITE (IOUT,260) TOTAL,SUM,XOLD(M),ZOLD(N)         NEW-1190
210         CONTINUE                                                       NEW-1200
220     CONTINUE                                                           NEW-1210
        TOTAL=UOLD(37)*(XOLD(45)+UOLD(23))                                 NEW-1220
        ALL=ALL+TOTAL                                                      NEW-1230
        IF (PR(22)) WRITE (IOUT,260) TOTAL,UOLD(37),XOLD(45),UOLD(23)      NEW-1240
        XNEW(49)=XNEW(49)+TEMP1+TEMP2+TEMP3+TEMP4+ALL+ZOLD(21)+ZOLD(22)    NEW-1250
        IF (PR(22)) WRITE (IOUT,280) XNEW(49)                              NEW-1260
C       DO SOME CLEANUP IF THIS IS A TWO - YEAR SECTOR                     NEW-1270
        IF (.NOT.TRANSF) RETURN                                            NEW-1280
        M=28                                                               NEW-1290
        DO 230 I=1,16                                                      NEW-1300
          M=M+1                                                            NEW-1310
          SAVE(I,K)=XNEW(I+28)                                             NEW-1320
230       XNEW(I+28)=0.0                                                   NEW-1330
        RETURN                                                             NEW-1340
C                                                                          NEW-1350
240     FORMAT (/17H NCB CONSTRUCTION,/,10X,F18.2,5X,39HBALANCE FROM CONSTNEW-1360
       1ANT COEFFICIENT TERMS,/,10X,F18.2,5X,26HRETURN ON RESTRICTED GIFTSNEW-1370
       2,/,10X,F18.2,5X,31HSTATE CONSTRUCTION FUNDS (LUMP),/,10X,F18.2,5X,NEW-1380
       333HFEDERAL CONSTRUCTION FUNDS (LUMP),/,10X,F18.2,5X,31HSTATE CONSTNEW-1390
       4RUCTION FUNDS (/ASF),/,10X,F18.2,5X,28HSTATE CONSTRUCTION AID (/LDNEW-1400
       5),/,10X,F18.2,5X,28HSTATE CONSTRUCTION AID (/UD),/,10X,F18.2,5X,30NEW-1410
       6HSTATE CONSTRUCTION AID (/GRAD),/,10X,F18.2,5X,31HFEDERAL CONSTRUCNEW-1420
       7TION AID (/ASF),/,10X,F18.2,5X,30HFEDERAL CONSTRUCTION AID (/LD),/NEW-1430
       8,10X,F18.2,5X,30HFEDERAL CONSTRUCTION AID (/UD),/,10X,F18.2,5X,32HNEW-1440
       9FEDERAL CONSTRUCTION AID (/GRAD),/,1X,F18.2,5X,5HTOTAL)            NEW-1450
250     FORMAT (/,14H NCB OPERATING,/,10X,F18.2,5X,39HBALANCE FROM CONSTANNEW-1460
       1T COEFFICIENT TERMS,/,10X,F18.2,5X,26HOTHER ADMINISTRATIVE COSTS,/NEW-1470
       2,10X,F18.2,5X,32HSTATE OPERATING FUNDS (/STUDENT),/,10X,F18.2,5X,1NEW-1480
       35HTUITION REVENUE,/,10X,F18.2,5X,35HFEDERAL AID TO INSTITUTIONS (INEW-1490
       4,A,L),/,10X,F18.2,5X,28HSTATE OPERATING FUNDS (LUMP),/,10X,F18.2,5NEW-1500
       5X,30HFEDERAL OPERATING FUNDS (LUMP))                               NEW-1510
260     FORMAT (10X,F18.2,20X,3F16.0)                                      NEW-1520
270     FORMAT (////,28H NEWX CALLED FOR TIME PERIOD,I4)                   NEW-1530
280     FORMAT (1X,F18.2,5X,5HTOTAL)                                       NEW-1540
        END                                                                NEW-1550
```

BEST COPY AVAILABLE


```
      SUBROUTINE CRITER (IPER)                                          CRI-0010
C                                                                       CRI-0020
C        XNEW IS THE VECTOR OF VALUES REL                               CRI-0030
C     IPER SPECIFIES THE TIME PERIOD INVOLVED                           CRI-0040
      REAL NHLAST,NHS                                                   CRI-0050
      LOGICAL TRANSF                                                    CRI-0060
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),          CRI-0070
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                         CRI-0080
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),     CRI-0090
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG    CRI-0100
      COMMON C(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                  CRI-0110
      COMMON TRAN(12,16)                                                CRI-0120
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                         CRI-0130
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                 CRI-0140
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           CRI-0150
      LOGICAL PR,TEST                                                   CRI-0160
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            CRI-0170
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           CRI-0180
      COMMON/NAMES/NUM(16)                                              CRI-0190
      REAL LD,LOSS(70),TERM(70)                                         CRI-0200
C                                                                       CRI-0210
C     CRITER WILL CALCULATE THE LOSS FUNCTION FOR ANY GIVEN TIME PERIOD CRI-0220
C     SPECIFIED BY *IPER*.  NOTICE THAT THIS ROUTINE, LIKE *DEBUG* AND  CRI-0230
C     UXDUMP*, IS NON-FUNCTIONAL IN THE SENSE THAT ONLY PRINTER OUTPUT  CRI-0240
C     IS GENERATED--NO NECESSARY CALCULATIONS ARE PERFORMED HERE.       CRI-0250
C                                                                       CRI-0260
C     FOUR PRINTER CONTROLS SPECIFY OUTPUT                              CRI-0270
C        3 - PRINTS SIMPLE STATISTICS                                   CRI-0280
C        4 - PRINTS THE LOSS FUNCTION FOR THIS PERIOD                   CRI-0290
C        19 - PRINTS AN ENTRY TRADER                                    CRI-0300
C        30 - PRINTS THE COMPOSITION OF THE LOSS FUNCTION BY TERM       CRI-0310
C                                                                       CRI-0320
C     THE INDEXED REPRESENTATION OF TERMS IN THE LOSS FUNCTION          CRI-0330
C     IS PROVIDED FOR DIAGNOSTIC PURPOSES ONLY.  THESE INDICES,HOWEVER, CRI-0340
C     CORRESPOND TO THE OBJECTIVE FUNCTION WEIGHTS AND TARGETS.         CRI-0350
C                                                                       CRI-0360
C                                                                       CRI-0370
      V(IPER)=0.0                                                       CRI-0380
      IF (PR(19)) WRITE (ICONO,150) IPER                                CRI-0390
      IF (IPER.EQ.LPP) RETURN                                           CRI-0400
C     CALCULATE GENERAL STATISTICS                                      CRI-0410
      LD=0.0                                                            CRI-0420
      UD=0.0                                                            CRI-0430
      DO 110 I=1,16                                                     CRI-0440
        LD=LD+X(I+12,IPER)+U(I+6,IPER)                                  CRI-0450
        UD=UD+X(I+28,IPER)+Z(I,IPER)                                    CRI-0460
110     CONTINUE                                                        CRI-0470
      GRAD=X(45,IPER)+U(23,IPER)                                        CRI-0480
      TS=LD+UD+GRAD                                                     CRI-0490
      F1=U(1,IPER)+X(1,IPER)                                            CRI-0500
      F2=U(2,IPER)+X(2,IPER)                                            CRI-0510
      F3=U(3,IPER)+X(3,IPER)                                            CRI-0520
```

BEST COPY AVAILABLE


```
      F4=U(4,IPER)+X(4,IPER)                                              CRI-0530
      TF=F1+F2+F3+F4                                                      CRI-0540
      TSP=X(8,IPER)+X(12,IPER)                                            CRI-0550
      BAL=X(48,IPER)+X(49,IPER)                                           CRI-0560
C     PRINT STATISTICS IF REQUESTED                                       CRI-0570
      IF (PR(3)) WRITE (IOUT,160) LD,UD,GRAD,TS,F1,F2,F3,F4,TF,TSP,BAL    CRI-0580
C     COMMENCE TERM BY TERM CALCULATION                                   CRI-0590
      TERM(1)=TS/TF                                                       CRI-0600
      TERM(2)=F1/TF                                                       CRI-0610
      TERM(3)=F2/TF                                                       CRI-0620
      TERM(4)=F3/TF                                                       CRI-0630
      TERM(5)=X(8,IPER)/TSP                                               CRI-0640
      TERM(6)=SPACE(1)+SPACE(2)*TS+SPACE(3)*TF                            CRI-0650
      R(6)=X(8,IPER)                                                      CRI-0660
      TERM(7)=SPACE(4)+SPACE(5)*TS+SPACE(6)*TF                            CRI-0670
      R(7)=X(12,IPER)                                                     CRI-0680
      TERM(8)=U(38,IPER)                                                  CRI-0690
      TERM(9)=X(47,IPER)                                                  CRI-0700
      TERM(10)=BAL                                                        CRI-0710
      TERM(11)=R(11)                                                      CRI-0720
      TERM(12)=LD/TS                                                      CRI-0730
      TERM(13)=UD/TS                                                      CRI-0740
      DO 120 I=14,29                                                      CRI-0750
120     TERM(I)=U(I-7,IPER)                                               CRI-0760
      TERM(30)=R(30)                                                      CRI-0770
      M=30                                                                CRI-0780
      DO 130 I=35,36                                                      CRI-0790
        TEMP1=U(I,IPER)                                                   CRI-0800
        DO 130 J=27,30                                                    CRI-0810
        TEMP2=TEMP1+U(J,IPER)                                             CRI-0820
        DO 130 K=31,34                                                    CRI-0830
        M=M+1                                                             CRI-0840
130     TERM(M)=TEMP2+U(K,IPER)                                           CRI-0850
      TERM(63)=U(37,IPER)                                                 CRI-0860
      TERM(64)=U(5,IPER)/X(8,IPER)                                        CRI-0870
      TERM(65)=U(6,IPER)/X(12,IPER)                                       CRI-0880
      TERM(66)=U(24,IPER)                                                 CRI-0890
      TERM(67)=U(25,IPER)                                                 CRI-0900
      TERM(68)=U(26,IPER)                                                 CRI-0910
      TEMP=0.0                                                            CRI-0920
      DO 140 I=1,NTERMS                                                   CRI-0930
        LOSS(I)=WT(I)*(TERM(I)-R(I))**2                                   CRI-0940
140     TEMP=TEMP+LOSS(I)                                                 CRI-0950
      V(IPER)=TEMP                                                        CRI-0960
      IF (PR(30)) WRITE (IOUT,170) (I,TERM(I),R(I),WT(I),LOSS(I),I=1,NTECRI-0970
     1RMS)                                                                CRI-0980
      IF (PR(4)) WRITE (IOUT,180) V(IPER)                                 CRI-0990
      RETURN                                                              CRI-1000
C                                                                         CRI-1010
C                                                                         CRI-1020
C                                                                         CRI-1030
150   FORMAT (////,38H ......ENTERING CRITER FOR TIME PERIOD,I3)          CRI-1040
```

BEST COPY AVAILABLE


```
160   FORMAT (///,11H STATISTICS,/,10X,5HLD  =,G12.4,/,10X,5HUD  =,G12.4CRI-1050
     1,/,10X,5HGRAD=,G12.4/,10X,5HTS  =,G12.4/,10X,5HF1  =,G12.4/,10X,5HCRI-1060
     2F2  =,G12.4/,10X,5HF3  =,G12.4/,10X,5HF4  =,G12.4/,10X,5HTF  =,G12CRI-1070
     3.4/,10X,5HTSP =,G12.4/,10X,5HBAL =,G12.4)                        CRI-1080
170   FORMAT (6X,54HINDEX    TERM         TARGET         WEIGHT          LOCRI-1090
     1SS/(19,4G14.4))                                                  CRI-1100
180   FORMAT (///,38H LOSS FUNCTION FOR THIS TIME PERIOD IS,G14.4)      CRI-1110
      END                                                              CRI-1120
```

BEST COPY AVAILABLE


```
      SUBROUTINE UPTRAN (K)                                                 UPT-0010
C     ON ENTRY K INDICATES APPROPRIATE TIME PERIOD.                         UPT-0020
C                                                                           UPT-0030
C     THE PURPOSE OF THIS ROUTINE IS TO UPDATE THE CURRENT F AND G          UPT-0040
C     MATRICES TO REFLECT CHANGES IN COEFFICIENTS WHICH ARE TIME-           UPT-0050
C     DEPENDENT.  NOTE THAT THESE TIME DEPENDENCIES OFTEN ARISE FROM        UPT-0060
C     NON-LINEARITIES IN THE TRANSITION EQUATIONS.  ALL CONSTANT            UPT-0070
C     COEFFICIENTS ARE ASSUMED TO RESIDE ALREADY IN F AND G.                UPT-0080
C     THE ONLY TRANSITION EQUATIONS WHICH REQUIRE UPDATING ARE THE          UPT-0090
C     NET CASH BALANCE EQUATIONS FOR CONSTRUCTION AND OPERATIONS (48        UPT-0100
C     AND 49).THE APPROACH TAKEN IS TO APPEND THESE CHANGES AT THE ENDS     UPT-0110
C     OF THE EXISTING F AND G MATRIX LINK STRUCTURES.  THUS, ONCE           UPT-0120
C     UPTRAN* HAS BEEN CALLED THE F AND G STRUCTURES FULLY REPRESENT THEUPT-0130
C     TRANSITION MATRICES USED IN MAYNES ALGORITHM.                         UPT-0140
C                                                                           UPT-0150
      REAL NHLAST,NHS                                                       UPT-0160
      LOGICAL TRANSF                                                        UPT-0170
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),              UPT-0180
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                            UPT-0190
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),         UPT-0200
     1  VALG(150),NUMF,NUMG,XT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG        UPT-0210
      COMMON C(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                      UPT-0220
      COMMON TRAN(12,16)                                                    UPT-0230
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                             UPT-0240
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                     UPT-0250
      LOGICAL PR,TEST                                                       UPT-0260
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,                UPT-0270
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                               UPT-0280
      COMMON/FILES/IIN,ICONI,ICOND,IOUT,ICUMP                               UPT-0290
      COMMON/NAMES/NUM(16)                                                  UPT-0300
C                                                                           UPT-0310
C                                                                           UPT-0320
C     FIRST COMPLETE NCB CONSTRUCTION FOR F MATRIX.                         UPT-0330
      TEMP=Q(2,K)+Q(6,K)                                                    UPT-0340
      LAST=NUMF+16                                                          UPT-0350
      NEXT=NUMF+1                                                           UPT-0360
      M=13                                                                  UPT-0370
      DO 110 I=NEXT,LAST                                                    UPT-0380
        INDIF(I)=48                                                         UPT-0390
        INDJF(I)=M                                                          UPT-0400
        VALF(I)=TEMP                                                        UPT-0410
110     M=M+1                                                               UPT-0420
      UEMP=Q(3,K)+Q(7,K)                                                    UPT-0430
      NEXT=LAST+1                                                           UPT-0440
      LAST=NEXT+15                                                          UPT-0450
      M=29                                                                  UPT-0460
      DO 120 I=NEXT,LAST                                                    UPT-0470
        INDIF(I)=48                                                         UPT-0480
        INDJF(I)=M                                                          UPT-0490
        VALF(I)=UEMP                                                        UPT-0500
120     M=M+1                                                               UPT-0510
      VEMP=Q(4,K)+Q(8,K)                                                    UPT-0520
```

BEST COPY AVAILABLE


```
      NF=LAST+1                                                          UPT-0530
      INDIF(NF)=48                                                       UPT-0540
      INDJF(NF)=45                                                       UPT-0550
      VALF(NF)=VEMP                                                      UPT-0560
C     COMPLETE NCB CONSTRUCTION FOR G MATRIX                             UPT-0570
C     NOTE..NEXT TWO SLOTS ALREADY HAVE ENTRIES IN CONSTANT PART         UPT-0580
      WEMP=Q(1,K)+Q(5,K)                                                 UPT-0590
      NEXT=NUMG+1                                                        UPT-0600
      INDIG(NEXT)=48                                                     UPT-0610
      INDJG(NEXT)=5                                                      UPT-0620
      VALG(NEXT)=WEMP                                                    UPT-0630
      NEXT=NEXT+1                                                        UPT-0640
      INDIG(NEXT)=48                                                     UPT-0650
      INDJG(NEXT)=6                                                      UPT-0660
      VALG(NEXT)=WEMP                                                    UPT-0670
      NEXT=NEXT+1                                                        UPT-0680
      LAST=NEXT+15                                                       UPT-0690
      M=7                                                                UPT-0700
      DO 130 I=NEXT,LAST                                                 UPT-0710
        INDIG(I)=48                                                      UPT-0720
        INDJG(I)=M                                                       UPT-0730
        VALG(I)=TEMP                                                     UPT-0740
130     M=M+1                                                            UPT-0750
      NG=LAST+1                                                          UPT-0760
      INDIG(NG)=48                                                       UPT-0770
      INDJG(NG)=23                                                       UPT-0780
      VALG(NG)=VEMP                                                      UPT-0790
C                                                                        UPT-0800
C     PROCEED WITH NCB OPERATING FOR F-MATRIX.                           UPT-0810
C                                                                        UPT-0820
      TEMP=U(38,K)+Q(10,K)+Q(9,K)                                        UPT-0830
      SEMP=TEMP+U(35,K)                                                  UPT-0840
      N=7                                                                UPT-0850
      M=13                                                               UPT-0860
      DO 150 I=27,30                                                     UPT-0870
        VEMP=SEMP+U(I,K)                                                 UPT-0880
        DO 140 IU=31,34                                                  UPT-0890
          NG=NG+1                                                        UPT-0900
          NF=NF+1                                                        UPT-0910
          INDIF(NF)=49                                                   UPT-0920
          INDJF(NF)=M                                                    UPT-0930
          INDIG(NG)=49                                                   UPT-0940
          INDJG(NG)=N                                                    UPT-0950
          SUM=VEMP+U(IU,K)+Q(M-2,K)                                      UPT-0960
          VALG(NG)=SUM                                                   UPT-0970
          VALF(NF)=SUM                                                   UPT-0980
          N=N+1                                                          UPT-0990
140       M=M+1                                                          UPT-1000
150     CONTINUE                                                         UPT-1010
      SEMP=TEMP+U(36,K)                                                  UPT-1020
      M=29                                                               UPT-1030
      DO 170 I=27,30                                                     UPT-1040
```

BEST COPY AVAILABLE


```
      VEMP=SEMP+U(I,K)                                                  UPT-1050
      DO 160 IU=31,34                                                   UPT-1060
        NF=NF+1                                                         UPT-1070
        INDIF(NF)=49                                                    UPT-1080
        INDJF(NF)=M                                                     UPT-1090
        VALF(NF)=VEMP+U(IU,K)+Q(M-2,K)                                  UPT-1100
160     M=M+1                                                           UPT-1110
170   CONTINUE                                                          UPT-1120
      TEMP=TEMP+Q(43,K)+U(37,K)                                         UPT-1130
      NF=NF+1                                                           UPT-1140
      NG=NG+1                                                           UPT-1150
      INDIG(NG)=49                                                      UPT-1160
      INDJG(NG)=23                                                      UPT-1170
      VALG(NG)=TEMP                                                     UPT-1180
      INDIF(NF)=49                                                      UPT-1190
      INDJF(NF)=45                                                      UPT-1200
      VALF(NF)=TEMP                                                     UPT-1210
C     COMPLETE NCB OPERATING FOR G-MATRIX                               UPT-1220
      M=27                                                              UPT-1230
      DO 190 I=13,25,4                                                  UPT-1240
        SUM=0.0                                                         UPT-1250
        NG=NG+1                                                         UPT-1260
        J=I+3                                                           UPT-1270
        DO 180 L=I,J                                                    UPT-1280
180       SUM=SUM+X(L,K)+X(L+16,K)+U(L-6,K)+Z(L-12,K)                   UPT-1290
        INDIG(NG)=49                                                    UPT-1300
        INDJG(NG)=M                                                     UPT-1310
        VALG(NG)=SUM                                                    UPT-1320
190     M=M+1                                                           UPT-1330
      M=31                                                              UPT-1340
      DO 210 I=13,16                                                    UPT-1350
        SUM=0.0                                                         UPT-1360
        NG=NG+1                                                         UPT-1370
        J=I+12                                                          UPT-1380
        DO 200 L=I,J,4                                                  UPT-1390
200       SUM=SUM+X(L,K)+X(L+16,K)+U(L-6,K)+Z(L-12,K)                   UPT-1400
        INDIG(NG)=49                                                    UPT-1410
        INDJG(NG)=M                                                     UPT-1420
        VALG(NG)=SUM                                                    UPT-1430
210     M=M+1                                                           UPT-1440
      NG=NG+1                                                           UPT-1450
      SUM=0.0                                                           UPT-1460
      TEMP=0.0                                                          UPT-1470
      DO 220 I=13,28                                                    UPT-1480
        TEMP=TEMP+X(I+16,K)+Z(I-12,K)                                   UPT-1490
220     SUM=SUM+X(I,K)+U(I-6,K)                                         UPT-1500
      INDIG(NG)=49                                                      UPT-1510
      INDJG(NG)=35                                                      UPT-1520
      VALG(NG)=SUM                                                      UPT-1530
      NG=NG+1                                                           UPT-1540
      INDIG(NG)=49                                                      UPT-1550
      INDJG(NG)=36                                                      UPT-1560
```

BEST COPY AVAILABLE


```
      VALG(NG)=TEMP                                                    UPT-1570
      NG=NG+1                                                          UPT-1580
      UEMP=X(45,K)+U(23,K)                                             UPT-1590
      INDIG(NG)=49                                                     UPT-1600
      INDJG(NG)=37                                                     UPT-1610
      VALG(NG)=UEMP                                                    UPT-1620
      NG=NG+1                                                          UPT-1630
      INDIG(NG)=49                                                     UPT-1640
      INDJG(NG)=38                                                     UPT-1650
      VALG(NG)=SUM+TEMP+UEMP                                           UPT-1660
      RETURN                                                           UPT-1670
      END                                                              UPT-1680
```

BEST COPY AVAILABLE


```
      SUBROUTINE DEBUG (IFLAG,II,AR)                                    DEB-0010
      DIMENSION AR(49)                                                  DEB-0020
C                                                                       DEB-0030
C        IFLAG = 0 MEANS USE DATA SUPPLIED--CONSTANT COEFFICIENTS       DEB-0040
C        IFLAG = 1 MEANS USE APPENDED COEFFICIENTS--PREVIOUSLY          DEB-0050
C                  CALCUALTED (NON-LINEARITIES)                         DEB-0060
C        IT = CURRENT TIME PERIOD                                       DEB-0070
C        AR = VECTOR OF LINEAR EQUATION VALUES FOR NEXT TIME PERIOD.    DEB-0080
C                                                                       DEB-0090
C                                                                       DEB-0100
C     THIS ROUTINE IS DESIGNED TO AID IN TRACING AND VERIFYING THE      DEB-0110
C     LINKED STRUCTURE OF THE F AND G MATRICES.  NO COMPUTATIONS ARE    DEB-0120
C     PERFORMED--ONLY LINEAR EQUATION-LIKE OUTPUT IS PRODUCED.  THE     DEB-0130
C     VALUE OF THE RESULTING LINEAR EQUATION FOR THE NEXT TIME PERIOD   DEB-0140
C     IS PRINTED, BUT THIS MAY OR MAY NOT BE MEANINGFUL INFO DEPENDING  DEB-0150
C     UPON THE CONTEXT IN WHICH *DEBUG* WAS CALLED.                     DEB-0160
C                                                                       DEB-0170
C                                                                       DEB-0180
      REAL NHLAST,NHS                                                   DEB-0190
      LOGICAL TRANSF                                                    DEB-0200
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),          DEB-0210
     1         Y(16),HS(16),ABAR(13),CHAT(16,13)                        DEB-0220
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),     DEB-0230
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG    DEB-0240
      COMMON Q(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                  DEB-0250
      COMMON TRAN(12,16)                                                DEB-0260
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                         DEB-0270
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                 DEB-0280
      LOGICAL PR,TEST                                                   DEB-0290
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            DEB-0300
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           DEB-0310
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           DEB-0320
      COMMON/NAMES/NUM(16)                                              DEB-0330
      IF (IFLAG.NE.0) GO TO 110                                         DEB-0340
      IFPTR=1                                                           DEB-0350
      IGPTR=1                                                           DEB-0360
      LIMITF=NUMF                                                       DEB-0370
      LIMITG=NUMG                                                       DEB-0380
      WRITE (IOUT,220)                                                  DEB-0390
      GO TO 120                                                         DEB-0400
110   IFPTR=NUMF+1                                                      DEB-0410
      IGPTR=NUMG+1                                                      DEB-0420
      LIMITF=NF                                                         DEB-0430
      LIMITG=NG                                                         DEB-0440
      WRITE (IOUT,230)                                                  DEB-0450
C                                                                       DEB-0460
120   LAST=INDIF(IFPTR)                                                 DEB-0470
      WRITE (IOUT,190) LAST,AR(LAST)                                    DEB-0480
130   K=INDJF(IFPTR)                                                    DEB-0490
      WRITE (IOUT,200) INDIF(IFPTR),K,K,VALF(IFPTR),X(K,IT)             DEB-0500
      IF (IFPTR.LT.LIMITF) GO TO 140                                    DEB-0510
      IF (IGPTR.LT.LIMITG) GO TO 150                                    DEB-0520
```

BEST COPY AVAILABLE


```
      RETURN                                                            DEB-0530
140   IFPTR=IFPTR+1                                                     DEB-0540
      IF (INDIF(IFPTR).EQ.LAST) GO TO 130                               DEB-0550
150   MAST=INDIG(IGPTR)                                                 DEB-0560
      IF (MAST.NE.LAST) GO TO 180                                       DEB-0570
160   K=INDJG(IGPTR)                                                    DEB-0580
      WRITE (IOUT,210) INDIG(IGPTR),K,K,VALG(IGPTR),U(K,1T)             DEB-0590
      IF (IGPTR.LT.LIMITG) GO TO 170                                    DEB-0600
      IF (IFPTR.LT.LIMITF) GO TO 120                                    DEB-0610
      RETURN                                                            DEB-0620
170   IGPTR=IGPTR+1                                                     DEB-0630
      IF (INDIG(IGPTR).EQ.MAST) GO TO 160                               DEB-0640
180   IF (INDIF(IFPTR).LE.INDIG(IGPTR)) GO TO 120                       DEB-0650
      MAST=INDIG(IGPTR)                                                 DEB-0660
      WRITE (IOUT,190) MAST,AR(LAST)                                    DEB-0670
      GO TO 160                                                         DEB-0680
C                                                                       DEB-0690
C                                                                       DEB-0700
C                                                                       DEB-0710
190   FORMAT (/5X,2HX(,I2,4H) = ,E15.4)                                 DEB-0720
200   FORMAT (10X,4H+ F(,I2,1H,,I2,6H) * X(,I2,1H),10X,G10.2,3H * ,G10.2DEB-0730
     1)                                                                 DEB-0740
210   FORMAT (10X,4H+ G(,I2,1H,,I2,6H) * U(,I2,1H),10X,G10.2,3H * ,G10.2DEB-0750
     1)                                                                 DEB-0760
220   FORMAT (///,49H TRANSITION  EQUATIONS WITH CONSTANT COEFFICIENTS,/DEB-0770
     1)                                                                 DEB-0780
230   FORMAT (///,48H TRANSITION EQUATIONS WITH VARIABLE COEFFICIENTS,/)DEB-0790
      END                                                               DEB-0800
```

BEST COPY AVAILABLE


```
      SUBROUTINE UXDUMP                                                 UXD-0010
C                                                                       UXD-0020
C     THE ONLY PURPOSE OF THIS ROUTINE IS TO SELECTIVELY DUMP           UXD-0030
C     THE CURRENT STATE AND CONTROL VARIABLES.                          UXD-0040
C                                                                       UXD-0050
      REAL NHLAST,NHS                                                   UXD-0060
      LOGICAL TRANSF                                                    UXD-0070
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),          UXD-0080
     1         Y(16),HS(16),ABAR(13),CHAT(16,13)                        UXD-0090
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),     UXD-0100
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG    UXD-0110
      COMMON G(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                  UXD-0120
      COMMON TRAN(12,16)                                                UXD-0130
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                         UXD-0140
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                 UXD-0150
      LOGICAL PR,TEST                                                   UXD-0160
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            UXD-0170
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           UXD-0180
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           UXD-0190
      COMMON/NAMES/NUM(16)                                              UXD-0200
C                                                                       UXD-0210
      WRITE (IOUT,130)                                                  UXD-0220
      WRITE (IOUT,140) (J,J=1,NPP)                                      UXD-0230
      DO 110 I=1,NU                                                     UXD-0240
110     WRITE (IOUT,170) I,(U(I,J),J=1,NPP)                             UXD-0250
      WRITE (IOUT,150)                                                  UXD-0260
      WRITE (IOUT,140) (J,J=1,LPP)                                      UXD-0270
      DO 120 I=1,NX                                                     UXD-0280
120     WRITE (IOUT,160) I,(X(I,J),J=1,LPP)                             UXD-0290
      RETURN                                                            UXD-0300
C                                                                       UXD-0310
C                                                                       UXD-0320
130   FORMAT (////,18H CONTROL VARIABLES,/)                             UXD-0330
140   FORMAT (/(3X,10I12))                                              UXD-0340
150   FORMAT (////,16H STATE VARIABLES,/)                               UXD-0350
160   FORMAT (2X,2HX(,I2,2H) ,10G12.4/(8X,10G12.4/))                    UXD-0360
170   FORMAT (2X,2HU(,I2,2H) ,8G12.4)                                   UXD-0370
      END                                                               UXD-0380
```

BEST COPY AVAILABLE


```
      SUBROUTINE CONTRL (IARG)                                          CON-0010
C                                                                       CON-0020
C        IARG = A CONTROL PARAMETER                                     CON-0030
C                                                                       CON-0040
C     THIS ROUTINE IS INTENDED AS THE USER-MODEL RUN-TIME INTERFACE.    CON-0050
C     ITS NATURE WILL VARY DEPENDING ON WHETHER THE MODEL IS USED IN    CON-0060
C     A BATCH OR INTERACTIVE ENVIRONMENT.                               CON-0070
C                                                                       CON-0080
C     THE FOLLOWING ABILITIES SHOULD BE PROVIDED.                       CON-0090
C        1) CONTROL NUMBER OF INSTITUTIONAL SECTORS TO BE RUN           CON-0100
C        2) SET OR RESET PRINTER SWITCHES                               CON-0110
C        3) SET OR RESET INSTITUTIONAL SECTOR ITERATION LIMITS          CON-0120
C                AND TOLERANCES                                         CON-0130
C                                                                       CON-0140
      LOGICAL PR,TEST                                                   CON-0150
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            CON-0160
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           CON-0170
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           CON-0180
      COMMON/NAMES/NUM(16)                                              CON-0190
      INTEGER COMAND(10)                                                CON-0200
      DATA (COMAND(I),I=1,6)/4HSTEP,4HRUNA,4HDROP,4HEDIT,4HSKIP,4HTEST/ CON-0210
      IF (IARG.NE.0) RETURN                                             CON-0220
      WRITE (ICONO,330)                                                 CON-0230
110   WRITE (ICONO,270)                                                 CON-0240
      READ (ICONI,280) INVAL                                            CON-0250
      DO 120 I=1,6                                                      CON-0260
        IF (INVAL.EQ.COMAND(I)) GO TO (140,150,160,170,220,230), I      CON-0270
120     CONTINUE                                                        CON-0280
130   WRITE (ICONO,290)                                                 CON-0290
      GO TO 110                                                         CON-0300
140   IARG=0                                                            CON-0310
      RETURN                                                            CON-0320
150   IARG=-1                                                           CON-0330
      RETURN                                                            CON-0340
160   WRITE (ICONO,300)                                                 CON-0350
      ENDFILE IOUT                                                      CON-0360
      STOP                                                              CON-0370
170   WRITE (ICONO,310)                                                 CON-0380
180   READ , INVALI                                                     CON-0390
      IF (INVALI.EQ.0) GO TO 110                                        CON-0400
      IF (INVALI.LT.0.OR.INVALI.GT.32) GO TO 210                        CON-0410
      IF (INVALI.EQ.31) GO TO 190                                       CON-0420
      IF (INVALI.EQ.32) GO TO 200                                       CON-0430
      PR(INVALI)=.NOT.PR(INVALI)                                        CON-0440
      WRITE (ICONO,240) INVALI                                          CON-0450
      GO TO 180                                                         CON-0460
190   WRITE (ICONO,250)                                                 CON-0470
      READ , EPSL                                                       CON-0480
      GO TO 180                                                         CON-0490
200   WRITE (ICONO,260)                                                 CON-0500
      READ , NUMIT                                                      CON-0510
      GO TO 180                                                         CON-0520
```

BEST COPY AVAILABLE


```
210   WRITE (ICONO,320)                                          CON-0530
      GO TO 180                                                  CON-0540
220   IARG=ISECT-1                                               CON-0550
      IF (IARG.EQ.0) GO TO 130                                   CON-0560
      NSECT=IARG                                                 CON-0570
      JSECT=IARG+1                                               CON-0580
      RETURN                                                     CON-0590
230   TEST=.TRUE.                                                CON-0600
      GO TO 110                                                  CON-0610
C                                                                CON-0620
C                                                                CON-0630
240   FORMAT (17H PRINTER CONTROL ,I2,10H SWITCHED.)             CON-0640
250   FORMAT (18H ENTER NEW EPSILON)                             CON-0650
260   FORMAT (26H ENTER NEW ITERATION LIMIT)                     CON-0660
270   FORMAT (22H ENTER CONTROL COMMAND)                         CON-0670
280   FORMAT (A4)                                                CON-0680
290   FORMAT (27H ILLEGAL COMMAND--TRY AGAIN)                    CON-0690
300   FORMAT (22H TERMINATION REQUESTED)                         CON-0700
310   FORMAT (34H ENTER CONTROL INDEX--0 TERMINATES)             CON-0710
320   FORMAT (21H BAD INDEX--TRY AGAIN)                          CON-0720
330   FORMAT (////)                                              CON-0730
      END                                                        CON-0740
```

BEST COPY AVAILABLE


```
      SUBROUTINE VIEWGO                                                 VIE-0010
C                                                                       VIE-0020
C                                                                       VIE-0030
      REAL NHLAST,NHS                                                   VIE-0040
      LOGICAL TRANSF                                                    VIE-0050
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),          VIE-0060
     1       Y(16),HS(16),ABAR(13),CHAT(16,13)                          VIE-0070
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),     VIE-0080
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG    VIE-0090
      COMMON C(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                  VIE-0100
      COMMON TRAN(12,16)                                                VIE-0110
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                         VIE-0120
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(4),49)                 VIE-0130
      LOGICAL PR,TEST                                                   VIE-0140
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            VIE-0150
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           VIE-0160
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           VIE-0170
      COMMON/NAMES/NUM(16)                                              VIE-0180
      COMMON/VUDAT/ENROL(6,17),DIF(6,17),FTSTU(6,17),STATS(6,6)         VIE-0190
      COMMON/DEMSUP/D(16,13),S(16,13)                                   VIE-0200
C                                                                       VIE-0210
      DIMENSION IORDR(6), SCALE(6)                                      VIE-0220
      REAL LD                                                           VIE-0230
      DATA IORDR/6,3,5,2,4,1/                                           VIE-0240
      DATA SCALE/6*1000.0/                                              VIE-0250
C                                                                       VIE-0260
C     THIS ROUTINE MERELY FORMATS DATA (COMMON BLOCK *VUDAT*) FOR USE   VIE-0270
C     BY THE OUTPUT REPORT GENERATOR *VIEW*.   NO CRITICAL              VIE-0280
C     COMPUTATIONS ARE PERFORMED IN THIS ROUTINE, AND ALL I/O WILL      VIE-0290
C     BE HANDLED BY *DATA*.                                             VIE-0300
C                                                                       VIE-0310
C                                                                       VIE-0320
      IF (PR(19)) WRITE (ICONO,240) ISECT                               VIE-0330
      IPER2=2                                                           VIE-0340
      II=IORDR(ISECT)                                                   VIE-0350
      M=12                                                              VIE-0360
      N=6                                                               VIE-0370
      TS=0.0                                                            VIE-0380
      SUM=0.0                                                           VIE-0390
      LD=0.0                                                            VIE-0400
      UD=0.0                                                            VIE-0410
      GRAD=0.0                                                          VIE-0420
      TFD=0.0                                                           VIE-0430
      DO 110 J=1,16                                                     VIE-0440
        N=N+1                                                           VIE-0450
        M=M+1                                                           VIE-0460
        T1=X(M,IPER2)+U(N,IPER2)                                        VIE-0470
        T2=X(M+16,IPER2)+Z(J,IPER2)                                     VIE-0480
      TFD=TFD+(VMS(J,IPER2)+VMS(J+33,IPER2))*T1                         VIE-0490
     1       +(VMS(J+16,IPER2)+VMS(J+49,IPER2))*T2                      VIE-0500
        LD=LD+T1                                                        VIE-0510
        UD=UD+T2                                                        VIE-0520
```

BEST COPY AVAILABLE


```
        ENROL(II,J)=T1+T2                                          VIE-0530
        TFD=TFD+Q(J+10,IPER2)*T1+Q(J+26,IPER2)*T2                  VIE-0540
        FTSTU(II,J)=U(N,IPER2)                                     VIE-0550
        DIF(II,J)=S(J,ISECT)-D(J,ISECT)                            VIE-0560
110     CONTINUE                                                   VIE-0570
      GRAD=X(45,IPER2)+U(23,IPER2)                                 VIE-0580
      ENROL(II,17)=GRAD                                            VIE-0590
      DIF(II,17)=0.0                                               VIE-0600
      TFD=TFD+GRAD*(VMS(33,IPER2)+VMS(66,IPER2))                   VIE-0610
      FTSTU(II,17)=U(23,IPER2)                                     VIE-0620
      TS=LD+UD+GRAD                                                VIE-0630
C     GET COST TERMS FROM THE CONSTANT COEFFICIENT                 VIE-0640
      DO 120 IF=1,150                                              VIE-0650
        IF (INDIF(IF).EQ.49) GO TO 130                             VIE-0660
120     CONTINUE                                                   VIE-0670
130   DO 140 IG=1,150                                              VIE-0680
        IF (INDIG(IG).EQ.49) GO TO 150                             VIE-0690
140     CONTINUE                                                   VIE-0700
150   CONTINUE                                                     VIE-0710
      DO 160 I=IF,NUMF                                             VIE-0720
        IF (VALF(I).GT.0.0) GO TO 160                              VIE-0730
        J=INDJF(I)                                                 VIE-0740
        SUM=SUM+VALF(I)*X(J,IPER2)                                 VIE-0750
160     CONTINUE                                                   VIE-0760
      DO 170 I=IG,NUMG                                             VIE-0770
        IF (VALG(I).GT.0.0) GO TO 170                              VIE-0780
        J=INDJG(I)                                                 VIE-0790
        SUM=SUM+VALG(I)*U(J,IPER2)                                 VIE-0800
170     CONTINUE                                                   VIE-0810
      SUM=SUM+Q(9,IPER2)*TS                                        VIE-0820
C     ADD IN STUDENT AND COSTS                                     VIE-0830
      M=12                                                         VIE-0840
      N=6                                                          VIE-0850
      DO 190 I=27,30                                               VIE-0860
        TEMP8=U(35,IPER2)+U(I,IPER2)                               VIE-0870
        DO 180 J=31,34                                             VIE-0880
          M=M+1                                                    VIE-0890
          N=N+1                                                    VIE-0900
          TEMP9=TEMP8+U(J,IPER2)                                   VIE-0910
          SUM=SUM+TEMP9*(X(M,IPER2)+U(N,IPER2))                    VIE-0920
180       CONTINUE                                                 VIE-0930
190     CONTINUE                                                   VIE-0940
      M=28                                                         VIE-0950
      N=0                                                          VIE-0960
      DO 210 I=27,30                                               VIE-0970
        TEMP8=U(36,IPER2)+U(I,IPER2)                               VIE-0980
        DO 200 J=31,34                                             VIE-0990
          M=M+1                                                    VIE-1000
          N=N+1                                                    VIE-1010
          TEMP9=TEMP8+U(J,IPER2)                                   VIE-1020
          SUM=SUM+TEMP9*(X(M,IPER2)+Z(N,IPER2))                    VIE-1030
200       CONTINUE                                                 VIE-1040
```

BEST COPY AVAILABLE


```
210     CONTINUE                                                        VIE-1050
C     FINALLY ADD IN STUDENT AID TO GRAD STUDENTS                       VIE-1060
      SUM=SUM+U(37,IPER2)*GRAD                                          VIE-1070
      SUM=-SUM                                                          VIE-1080
C     CALCULATE TOTAL FACULTY                                           VIE-1090
      TF=0.0                                                            VIE-1100
      DO 220 I=1,4                                                      VIE-1110
220     TF=TF+X(I,IPER2)+U(I,IPER2)                                     VIE-1120
C     CALCULATE TOTAL SPACE                                             VIE-1130
      TSP=X(8,IPER2)+X(12,IPER2)                                        VIE-1140
C     CALCULATE COST PER STUDENT                                        VIE-1150
      CPS=SUM/TS                                                        VIE-1160
C     CALCULATE TOTAL FEDERAL DOLLARS                                   VIE-1170
      TFD=TFD+Z(20,IPER2)+Z(22,IPER2)+Q(5,IPER2)*(X(8,IPER2)+X(12,IPERVIE-1180
     1  2))                                                            VIE-1190
      TFD=TFD+Q(6,IPER2)*LD+Q(7,IPER2)*UD+Q(8,IPER2)*GRAD+Q(45,IPER2)*GRVIE-1200
     1AD                                                              VIE-1210
C     ADD ON FED AID TO INST BY STUDENT GROUPS                          VIE-1220
C     FINALLY SET UP INSTITUTIONAL STATS                                VIE-1230
      STATS(II,1)=X(49,IPER2)/SCALE(1)                                  VIE-1240
      STATS(II,2)=TF/SCALE(2)                                           VIE-1250
      STATS(II,3)=TSP/SCALE(3)                                          VIE-1260
      STATS(II,4)=TS/SCALE(4)                                           VIE-1270
      STATS(II,5)=TFD/SCALE(5)                                          VIE-1280
      STATS(II,6)=CPS/SCALE(6)                                          VIE-1290
      RETURN                                                            VIE-1300
C                                                                       VIE-1310
240   FORMAT (////34H ......ENTERING VIEWGO FOR SECTOR ,I3)             VIE-1320
      END                                                               VIE-1330
```

BEST COPY AVAILABLE


```
      SUBROUTINE DATA (IVAL)                                              DAT-0010
C                                                                         DAT-0020
C     IVAL JUST SPECIFIES THE ACTION TO BE PERFORMED                      DAT-0030
C                                                                         DAT-0040
C     DATA IS DESIGNED AS A COLLECTION OF ALL NON-TRIVIAL HANDLING        DAT-0050
C     OF THE DATA BASE.  DATA FUNCTIONS ARE QUITE SIMPLE FOR              DAT-0060
C     FULLY RANDOM FILE STRUCTURES--HOWEVER DATA CAN BE HANDLED           DAT-0070
C     WITH ONE OR MORE SEQUENTIAL DEVICES.                                DAT-0080
C                                                                         DAT-0090
C                                                                         DAT-0100
      REAL NHLAST,NHS                                                     DAT-0110
      LOGICAL TRANSF                                                      DAT-0120
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),            DAT-0130
     1         Y(16),ES(16),ABAR(13),CHAT(16,13)                          DAT-0140
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),       DAT-0150
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG      DAT-0160
      COMMON G(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                    DAT-0170
      COMMON TRAN(12,16)                                                  DAT-0180
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                           DAT-0190
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                   DAT-0200
      LOGICAL PR,TEST                                                     DAT-0210
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,              DAT-0220
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                             DAT-0230
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,ICOMP                             DAT-0240
      COMMON/NAMES/NUM(16)                                                DAT-0250
      COMMON/VUDAT/ENROL(6,17),DIF(6,17),FTSTU(6,17),STATS(6,6)           DAT-0260
C                                                                         DAT-0270
C                                                                         DAT-0280
      DIMENSION INDEX(200), LENGTH(70), IRSECT(12)                        DAT-0290
      DIMENSION INDEX2(40)                                                DAT-0300
      DIMENSION NDICT(20,10)                                              DAT-0310
      EQUIVALENCE (NDICT,W)                                               DAT-0320
      DIMENSION BETA(38,49)                                               DAT-0330
      EQUIVALENCE (W,BETA)                                                DAT-0340
      DATA LENGTH/67,5*1862,60*0,4*192/                                   DAT-0350
      DATA IRSECT/67,68,69,9*0/                                           DAT-0360
      DATA ICHARI/1HI/                                                    DAT-0370
C                                                                         DAT-0380
      IF (PR(19)) WRITE (ICONO,360) IVAL                                  DAT-0390
      GO TO (110,140,150,160,190,200,210,220,320,330,340,350), IVAL       DAT-0400
C                                                                         DAT-0410
C     (1) IS CALLED TO INITIALIZE GLOBAL DATA VALUES WHICH (CURRENTLY)    DAT-0420
C     ARE BOTH TIME AND SECTOR INDEPENDENT.  IN 6000 IMPLEMENTATION,      DAT-0430
C     THIS CALL ALSO INITIALIZES MASS STORAGE.  LENGTH MERELY CONTAINS    DAT-0440
C     THE NUMBER OF 6000 WORDS IN EACH RECORD                             DAT-0450
110   CONTINUE                                                            DAT-0460
      WRITE (ICONO,410)                                                   DAT-0470
      READ (ICONI,380) INVAL                                              DAT-0480
      IF (INVAL.EQ.ICHARI) GO TO 120                                      DAT-0490
      PR(21)=.TRUE.                                                       DAT-0500
      IOUT=ICONO                                                          DAT-0510
120   CONTINUE                                                            DAT-0520
```

BEST COPY AVAILABLE


```
      CALL OPENMS (IDUMP,INDEX2,40,0)                                    DAT-0530
      CALL READMS (IDUMP,NDICT,342,1)                                    DAT-0540
      WRITE (ICONC,370)                                                  DAT-0550
      READ , KEX                                                         DAT-0560
      WRITE (ICONO,390)                                                  DAT-0570
      READ (ICONI,400) (NDICT(KEX,J),J=1,10)                             DAT-0580
      CALL WRITMS (IDUMP,NDICT,342,1,0)                                  DAT-0590
      CALL OPENMS (IIN,INDEX,200,0)                                      DAT-0600
      DO 130 I=7,62,5                                                    DAT-0610
        LENGTH(I)=1061                                                   DAT-0620
        LENGTH(I+1)=750                                                  DAT-0630
        LENGTH(I+2)=484                                                  DAT-0640
        LENGTH(I+3)=150                                                  DAT-0650
130     LENGTH(I+4)=150                                                  DAT-0660
      CALL READMS (IIN,NHLAST,LENGTH(1),1)                               DAT-0670
      RETURN                                                             DAT-0680
C                                                                        DAT-0690
C     (2) IS CALLED TO READ IN ALL SECTOR DEPENDENT INFO.  KEY SPECIFIESDAT-0700
C     THE TARGET RECORD AND IS DERIVED FROM ISECT                        DAT-0710
140   CONTINUE                                                           DAT-0720
      KEY=2+5*ISECT                                                      DAT-0730
      CALL READMS (IIN,INDIF,LENGTH(KEY),KEY)                            DAT-0740
      CALL READMS (IIN,Q,LENGTH(KEY+1),KEY+1)                            DAT-0750
      CALL READMS (IIN,U,LENGTH(KEY+2),KEY+2)                            DAT-0760
      RETURN                                                             DAT-0770
C                                                                        DAT-0780
C     (3) CAUSES THE NEWLY CALCULATED X AND U MATRICES TO                DAT-0790
C     BE WRITTEN TO DISK                                                 DAT-0800
150   CONTINUE                                                           DAT-0810
      KEY=4+5*ISECT                                                      DAT-0820
      CALL WRITMS (IIN,U,LENGTH(KEY),KEY,0)                              DAT-0830
      RETURN                                                             DAT-0840
C                                                                        DAT-0850
C  (4) CALLS FOR PREPARATION OF DATA FOR NEXT FEDERAL PLANNING PERIOD.   DAT-0860
C     THIS IS LIMITED TO NHS,X,U,Z.                                      DAT-0870
160   CONTINUE                                                           DAT-0880
      IF (IYR.NE.1) GO TO 170                                            DAT-0890
      CALL WRITMS (IDUMP,ENROL,342,KEX+1,0)                              DAT-0900
      CALL CLOSMS (IDUMP,INDEX2,40)                                      DAT-0910
170   CONTINUE                                                           DAT-0920
      NHLAST=NHS(1)                                                      DAT-0930
      DO 180 I=1,9                                                       DAT-0940
180     NHS(I)=NHS(I+1)                                                  DAT-0950
      RETURN                                                             DAT-0960
C                                                                        DAT-0970
C     (5) IS CALLED TO POSITION THE BETA PUSH-DOWN STACK.                DAT-0980
190   CONTINUE                                                           DAT-0990
      P=1                                                                DAT-1000
      RETURN                                                             DAT-1010
C                                                                        DAT-1020
C     (6) IS CALLED TO REPOSITION SEQUENTIAL FILES--NOT NEEDED HERE.     DAT-1030
200   CONTINUE                                                           DAT-1040
```

BEST COPY AVAILABLE


```
      RETURN                                                            DAT-1050
C                                                                       DAT-1060
C     (7) HAS THE SAME FUNCTION AS (2)                                  DAT-1070
210   CONTINUE                                                          DAT-1080
      GO TO 140                                                         DAT-1090
C                                                                       DAT-1100
C     (8) PROVIDES AN UPDATE OF VARIABLES FOR NEST FEDERAL PERIOD       DAT-1110
C     RESET UPDATED U AND X VALUES BEFORE WRITING TO DISK               DAT-1120
C     STASH X,Y AT END OF STUDENT SECTOR                                DAT-1130
220   CONTINUE                                                          DAT-1140
      IF (IYR.EQ.3) RETURN                                              DAT-1150
      DO 230 I=1,NX                                                     DAT-1160
230     X(I,1)=X(I,2)                                                   DAT-1170
      DO 240 I=1,NU                                                     DAT-1180
240     U(I,1)=U(I,2)                                                   DAT-1190
      KEY=4+ISECT*5                                                     DAT-1200
      CALL WRITMS (IIN,U(1,1),LENGTH(KEY),KEY,0)                        DAT-1210
      NPPM1=NPP-1                                                       DAT-1220
      DO 280 J=1,NPPM1                                                  DAT-1230
        DO 250 I=1,NV                                                   DAT-1240
250       VMS(I,J)=VMS(I+1,J)                                           DAT-1250
        DO 260 I=1,NQ                                                   DAT-1260
260       Q(I,J)=Q(I+1,J)                                               DAT-1270
        DO 270 I=1,NZ                                                   DAT-1280
270       Q(I,J)=Q(I+1,J)                                               DAT-1290
280     CONTINUE                                                        DAT-1300
      KEY1=KEY+IYR                                                      DAT-1310
      CALL READMS (IIN,VALF,LENGTH(KEY1),KEY1)                          DAT-1320
      DO 290 I=1,45                                                     DAT-1330
290     Q(I,NPP)=VALF(I)                                                DAT-1340
      DO 300 J=1,80                                                     DAT-1350
        I=I+1                                                           DAT-1360
300     VMS(J,NPP)=VALF(I)                                              DAT-1370
      DO 310 J=1,25                                                     DAT-1380
        I=I+1                                                           DAT-1390
310     VMS(J,NPP)=VALF(I)                                              DAT-1400
      KEY2=KEY-1                                                        DAT-1410
      CALL WRITMS (IIN,Q(1,1),LENGTH(KEY2),KEY2,0)                      DAT-1420
      RETURN                                                            DAT-1430
C                                                                       DAT-1440
C     (9) MERELY READS TRAN ON DEMAND                                   DAT-1450
320   CONTINUE                                                          DAT-1460
      KEY=IRSECT(ISECT)                                                 DAT-1470
      CALL READMS (IIN,TRAN,LENGTH(KEY),KEY)                            DAT-1480
      RETURN                                                            DAT-1490
C                                                                       DAT-1500
C     (10) IS DEFUNCT                                                   DAT-1510
330   CONTINUE                                                          DAT-1520
      RETURN                                                            DAT-1530
C                                                                       DAT-1540
C     (11) IS USED TO PUSH THE BETA STACK AND WRITE                     DAT-1550
340   CONTINUE                                                          DAT-1560
```

BEST COPY AVAILABLE


```
      M=M+1                                                              DAT-1570
      CALL WRITMS (IIN,BETA,LENGTH(M),M,0)                               DAT-1580
      RETURN                                                             DAT-1590
C                                                                        DAT-1600
C     (12) IS USED TO READ THE BETA STACK AND POP                        DAT-1610
320   CONTINUE                                                           DAT-1620
      CALL READMS (IIN,BETA,LENGTH(M),M)                                 DAT-1630
      M=M-1                                                              DAT-1640
      RETURN                                                             DAT-1650
C                                                                        DAT-1660
C                                                                        DAT-1670
360   FORMAT (23H .......DATA CALLED WITH,I3)                            DAT-1680
370   FORMAT (41H ENTER INDEX OF OUTPUT FILE TO BE CREATED)              DAT-1690
380   FORMAT (A1)                                                        DAT-1700
390   FORMAT (27H ENTER NAME FOR INDEXED RUN)                            DAT-1710
400   FORMAT (10A6)                                                      DAT-1720
410   FORMAT (1H1,10X,37HNATIONAL PLANNING MODEL - VERSION 2.4,////,34H  DAT-1730
     1BATCH OR INTERACTIVE JOB (B OR I))                                 DAT-1740
      END                                                                DAT-1750
```

BEST COPY AVAILABLE


```
      BLOCK DATA                                                        BLD-0010
      REAL NHLAST,NHS                                                   BLD-0020
      LOGICAL TRANSF                                                    BLD-0030
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),          BLD-0040
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                         BLD-0050
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),     BLD-0060
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG    BLD-0070
      COMMON Q(45,5),VMS(90,5),Z(25,5),U(38,5),X(49,6)                  BLD-0080
      COMMON TRAN(12,16)                                                BLD-0090
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                         BLD-0100
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                 BLD-0110
      COMMON/NAMES/NUM(16)                                              BLD-0120
      LOGICAL PR,TEST                                                   BLD-0130
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,            BLD-0140
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                           BLD-0150
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                           BLD-0160
      DATA TEST/.FALSE./                                                BLD-0170
      DATA NTERMS,NH/68,2/                                              BLD-0180
      DATA NX,NU,NZ,NV,NQ,NSECT,NPP/49,38,22,67,43,12,5/                BLD-0190
      DATA NUM/1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16/                  BLD-0200
      DATA EPSL,NUMIT,PR/0.01,4,30*.TRUE./                              BLD-0210
      DATA IIN,ICONI,ICONO,IOUT,IDUMP/1,5,6,42,2/                       BLD-0220
      END                                                               BLD-0230
```

BEST COPY AVAILABLE


```
      SUBROUTINE DSINV(A,N,EPS,IER)                                      DSI-0010
C ******************************************************************** DSI-0020
C                                                                        DSI-0030
C     PURPOSE                                                            DSI-0040
C      INVERT A GIVEN SYMMETRIC POSITIVE DEFINITE MATRIX                 DSI-0050
C                                                                        DSI-0060
C     USAGE                                                              DSI-0070
C      CALL DSINV(A,N,EPS,IER)                                           DSI-0080
C                                                                        DSI-0090
C     DESCRIPTION OF PARAMETERS                                          DSI-0100
C      A    DOUBLE PRECISION UPPER TRIANGULAR PART OF GIVEN SYMMETRIC    DSI-0110
C           POSITIVE DEFINITE N BY N MATRIX.                             DSI-0120
C           ON RETURN A CONTAINS THE RESULTANT UPPERTRIANGULAR MATRIX    DSI-0130
C           IN DOUBLE PRECISION.                                         DSI-0140
C      N    ORDER OF THE GIVEN MATRIX                                    DSI-0150
C      EPS  SINGLE PRECISION INPUT PARAMETER WHICH IS USED AS RELATIVE   DSI-0160
C           TOLERANCE FOR TEST ON LOSS OF SIGNIFICANCE.                  DSI-0170
C      IER  RESULTING ERROR PARAMETER CODED AS FOLLOWS                   DSI-0180
C           IER=0    NO ERROR                                           DSI-0190
C           IER=-1   NO RESULT BECAUSE OF WRONG INPUT PARAMETER N OR     DSI-0200
C                    BECAUSE SOME RADICAND IS NON-POSITIVE (MATRIX A IS  DSI-0210
C                    IS NOT POTITIVE DEFINITE, POSSIBLY DUE TO LOSS OF   DSI-0220
C                    SIGNIFICANCE,)                                      DSI-0230
C           IER=>0   WARNING WHICH INDICATES LOSS OF SIGNIFICANCE.       DSI-0240
C                                                                        DSI-0250
C     REMARKS                                                            DSI-0260
C      THE UPPER TRIANGULAR PART OF GIVEN MATRIX IS ASSUMED TO BE STORED DSI-0270
C      COLUMNWISE IN N*(N+1)/2 SUCCESSIVE STORAGE LOCATIONS.  IN THE     DSI-0280
C      SAME STORAGE LOCATIONS THE RESULTING UPPER TRIANGULAR MATRIX IS   DSI-0290
C      STORED COLUMNWISE TOO.                                            DSI-0300
C      THE PROCEDURE GIVES RESULTS IF N IS GREATER THAN 0 AND ALL        DSI-0310
C      CALCULATED RADICANDS ARE POSITIVE.                                DSI-0320
C                                                                        DSI-0330
C     SUBROUTINE REQUIRED - DMFSD                                        DSI-0340
C                                                                        DSI-0350
C ******************************************************************** DSI-0360
C                                                                        DSI-0370
      DIMENSION A(1)                                                     DSI-0380
C                                                                        DSI-0390
C        FACTORIZE GIVEN MATRIX BY MEANS OF SUBROUTINE DMFSD             DSI-0400
C        A=TRANSPOSE(T) * T                                              DSI-0410
      CALL DMFSD(A,N,EPS,IER)                                            DSI-0420
      IF(IER) 9,1,1                                                      DSI-0430
C                                                                        DSI-0440
C        INVERT UPPER TRIANGULAR MATRIX T                                DSI-0450
C        PREPARE INVERSION-LOOP                                          DSI-0460
    1 IPIV=N*(N+1)/2                                                     DSI-0470
      IND=IPIV                                                           DSI-0480
C                                                                        DSI-0490
C        INITIALIZE INVERSION-LOOP                                       DSI-0500
      DO 8 I=1,N                                                         DSI-0510
      DIN=1.0D0/A(IPIV)                                                  DSI-0520
```

BEST COPY AVAILABLE


```
      A(IPIV)=DIN                                                       DSI-0530
      MIN=N                                                             DSI-0540
      KEND=I-1                                                          DSI-0550
      LANF=N-KEND                                                       DSI-0560
      IF(KEND)5,5,2                                                     DSI-0570
    2 J=IND                                                             DSI-0580
C                                                                       DSI-0590
C        INITIALIZE ROW-LOOP                                            DSI-0600
      DO 4 K=1,KEND                                                     DSI-0610
      WORK=0.CO                                                         DSI-0620
      MIN=MIN-1                                                         DSI-0630
      LHOR=IPIV                                                         DSI-0640
      LVER=J                                                            DSI-0650
C                                                                       DSI-0660
C        START INNER LOOP                                               DSI-0670
      DO 3 L=LANF,MIN                                                   DSI-0680
      LVER=LVER+1                                                       DSI-0690
      LHOR=LHOR+L                                                       DSI-0700
    3 WORK=WORK+A(LVER)*A(LHOR)                                         DSI-0710
C        END OF INNER LOOP                                              DSI-0720
C                                                                       DSI-0730
      A(J)=-WORK*DIN                                                    DSI-0740
    4 J=J-MIN                                                           DSI-0750
C        END OF ROW-LOOP                                                DSI-0760
C                                                                       DSI-0770
    5 IPIV=IPIV-MIN                                                     DSI-0780
    6 IND=IND-1                                                         DSI-0790
C        END OF INVERSION-LOOP                                          DSI-0800
C                                                                       DSI-0810
C        CALCULATE INVERSE(A) BY MEANS OF INVERSE(T)                    DSI-0820
C        INVERSE(A)=INVERSE(T)*TRANSPOSE(INVERSE(T))                    DSI-0830
C        INITIALIZE MULTIPLICATION-LOOP                                 DSI-0840
      DO 8 I=1,N                                                        DSI-0850
      IPIV=IPIV+I                                                       DSI-0860
      J=IPIV                                                            DSI-0870
C                                                                       DSI-0880
C        INITIALIZE ROW-LOOP                                            DSI-0890
      DO 8 K=I,N                                                        DSI-0900
      WORK=0.CO                                                         DSI-0910
      LHOR=J                                                            DSI-0920
C                                                                       DSI-0930
C        START INNER LOOP                                               DSI-0940
      DO 7 L=K,N                                                        DSI-0950
      LVER=LHOR+K-I                                                     DSI-0960
      WORK=WORK+A(LHOR)*A(LVER)                                         DSI-0970
    7 LHOR=LHOR+L                                                       DSI-0980
C        END OF INNER LOOP                                              DSI-0990
C                                                                       DSI-1000
      A(J)=WORK                                                         DSI-1010
    8 J=J+K                                                             DSI-1020
C        END OF ROW- AND MULTIPLICATION-LOOP                            DSI-1030
C                                                                       DSI-1040
```

BEST COPY AVAILABLE


```
    9 RETURN                                                    DSI-1050
      END                                                       DSI-1060
```

BEST COPY AVAILABLE


```
      SUBROUTINE DMFSD(A,N,EPS,IER)                                      DMF-0010
C *************************************************************************DMF-0020
C                                                                          DMF-0030
C     PURPOSE                                                              DMF-0040
C      FACTOR A GIVEN SYMMETRIC POSITIVE DEFINITE MATRIX                   DMF-0050
C                                                                          DMF-0060
C     USAGE                                                                DMF-0070
C      CALL DMFSD(A,N,EPS,IER)                                             DMF-0080
C                                                                          DMF-0090
C     DESCRIPTION OF PARAMETERS                                            DMF-0100
C      A    DOUBLE PRECISION UPPER TRIANGULAR PART OF GIVEN                DMF-0110
C           SYMMETRIC POSITIVE DEFINITE N BY N MATRIX                      DMF-0120
C      N    THE NUMBER OF ROWS(COLUMNS) IN GIVEN MATRIX                    DMF-0130
C           ON RETURN A CONTAINS THE RESULTANT UPPER                       DMF-0140
C           TRIANGULAR MATRIX IN DOUBLE PRECISION                          DMF-0150
C      EPS  SINGLE PRECISION INPUT CONSTANT WHICH IS USED                  DMF-0160
C           AS RELATIVE TOLERANCE FOR TEST ON LOSS OF                      DMF-0170
C           SIGNIFICANCE                                                   DMF-0180
C      IER  RESULTING ERRORPARAMETER CODED AS FOLLOWS                      DMF-0190
C           IER=0 - NO ERROR                                               DMF-0200
C           IER=-1 - NO RESULT BECAUSE OF WRONG INPUT PARAMETER            DMF-0210
C                    N OR BECAUSE SOME RADICAND IS NON-POSITIVE            DMF-0220
C                    (MATRIX A IS NOT POSITIVEDEFINITE -                   DMF-0230
C                    POSSIBLY DUE TO LOSS OF SIGNIFICANCE)                 DMF-0240
C           IER=K  - WARNING WHICH INDICATES LOSS OF SIGNIFICANCE          DMF-0250
C                    THE RADICAND FORMED AT FACTORIZATION                  DMF-0260
C                    STEP K+1 WAS STILL POSITIVE BUT NO LONGER             DMF-0270
C                    GREATER THAN ABS(EPS*K+1,K+1))                        DMF-0280
C     REMARKS                                                              DMF-0290
C      THE UPPER TRIANGULAR PART OF GIVEN MATRIX IS ASSUMED TO BE          DMF-0300
C      STORED COLUMNWISE IN N*(N+1)/2 SUCCESSIVE STORAGE LOCATIONS.        DMF-0310
C      IN THE SAME STORAGE LOCATIONS THE RESULTING UPPER TRIANGULAR        DMF-0320
C      MATRIX IS STORED COLUMNWISE TOO.                                    DMF-0330
C      THE PROCEDURE GIVES RESULTS IF N IS GREATER THAN 0 AND ALL          DMF-0340
C      CALCULATED RADICANDS ARE POSITIVE.                                  DMF-0350
C      THE PRODUCT OF RETURNED DIAGONAL TERMS IS EQUAL TO THE              DMF-0360
C      SQUARE-ROOT OF THE DETERMINANT OF THE GIVEN MATRIX.                 DMF-0370
C                                                                          DMF-0380
C     METHOD                                                               DMF-0390
C      SOLUTION IS DONE USING THE SQUARE-ROOT METHOD OF CHOLESKY.          DMF-0400
C      THE GIVEN MATRIX IS REPRESENTED AS PRODUCT OF TWO TRIANGULAR        DMF-0410
C      MATRICES. THE LEFT HAND FACTOR IS THE TRANSPOSE OF THE              DMF-0420
C      THE RETURNED RIGHT HAND FACTOR.                                     DMF-0430
C                                                                          DMF-0440
C *************************************************************************DMF-0450
C                                                                          DMF-0460
C                                                                          DMF-0470
      DIMENSION A(1)                                                       DMF-0480
C                                                                          DMF-0490
C        TEST ON WRONG INPUT PARAMETER N                                   DMF-0500
      IF(N-1)12,1,1                                                        DMF-0510
    1 IER=0                                                                DMF-0520
```

BEST COPY AVAILABLE


```
C                                                                       DMF-0530
C        INITIALIZE DIAGNONAL-LOOP                                      DMF-0540
      KPIV=0                                                            DMF-0550
      DO 11 K=1,N                                                       DMF-0560
      KPIV=KPIV+K                                                       DMF-0570
      IND=KPIV                                                          DMF-0580
      LEND=K-1                                                          DMF-0590
C                                                                       DMF-0600
C        CALCULATE TOLERANCE                                            DMF-0610
      TOL=ABS(EPS*A(KPIV))                                              DMF-0620
C                                                                       DMF-0630
C        START FACTORIZATION-LOOP OVER K-TH ROW                         DMF-0640
      DO 11 I=K,N                                                       DMF-0650
      DSUM=0.D0                                                         DMF-0660
      IF(LEND)2,4,2                                                     DMF-0670
C                                                                       DMF-0680
C        START INNER LOOP                                               DMF-0690
    2 DO 3 L=1,LEND                                                     DMF-0700
      LANF=KPIV-L                                                       DMF-0710
      LIND=IND-L                                                        DMF-0720
    3 DSUM=DSUM+A(LANF)*A(LIND)                                         DMF-0730
C        END OF INNER LOOP                                              DMF-0740
C                                                                       DMF-0750
C        TRANSFORM ELEMENT A(IND)                                       DMF-0760
    4 DSUM=A(IND)-DSUM                                                  DMF-0770
      IF(I-K) 10,5,10                                                   DMF-0780
C                                                                       DMF-0790
C        TEST FOR NEGATIVE PIVOT ELEMENT AND FOR LOSS OF SIGNIFICANCE   DMF-0800
    5 IF(DSUM-TOL)6,6,9                                                 DMF-0810
    6 IF(DSUM) 12,12,7                                                  DMF-0820
    7 IF(IER) 8,8,9                                                     DMF-0830
    8 IER=K-1                                                           DMF-0840
C                                                                       DMF-0850
C        COMPUTE PIVOT ELEMENT                                          DMF-0860
    9 DPIV=SQRT(DSUM)                                                   DMF-0870
      A(KPIV)=DPIV                                                      DMF-0880
      DPIV=1.D0/DPIV                                                    DMF-0890
      GO TO 11                                                          DMF-0900
C                                                                       DMF-0910
C        CALCULATE TERMS IN ROW                                         DMF-0920
   10 A(IND)=DSUM*DPIV                                                  DMF-0930
   11 IND=IND+I                                                         DMF-0940
C        END OF DIAGONAL LOOP                                           DMF-0950
C                                                                       DMF-0960
      RETURN                                                            DMF-0970
   12 IER=-1                                                            DMF-0980
      RETURN                                                            DMF-0990
      END                                                               DMF-1000
```

BEST COPY AVAILABLE


```
      SUBROUTINE CALCV(IPER)                                                CAL-0010
      REAL NHLAST,NHS                                                       CAL-0020
      LOGICAL TRANSF                                                        CAL-0030
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),              CAL-0040
     1        Y(16),HS(16),ABAR(13),CHAT(16,13)                            CAL-0050
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),         CAL-0060
     1  VALG(150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG        CAL-0070
      COMMON G(45,5),VMS(80,5),Z(25,5),U(38,5),X(49,6)                      CAL-0080
      COMMON TRAN(12,16)                                                    CAL-0090
      COMMON VXX(49,49),VUX(38,49),VXX2(49,49),                             CAL-0100
     1   VX(49),V(6),VU(38),VX2(49),VUU(38,38),W(49,49)                     CAL-0110
      COMMON/FILES/IIN,ICONI,ICONC,IOUT,IDUMP                               CAL-0120
      LOGICAL PR,TEST                                                       CAL-0130
      COMMON/CVALS/EPSL,NUMIT,PR(30),NX,NU,NZ,NV,NQ,NPP,LPP,                CAL-0140
     1   NSECT,JSECT,ISECT,IYR,NTERMS,NH,TEST                               CAL-0150
      DIMENSION TU(10,10)                                                   CAL-0160
      DIMENSION SU(33)                                                      CAL-0170
      DIMENSION VXLIN(1)                                                    CAL-0180
      EQUIVALENCE (VXLIN,VXX2)                                              CAL-0190
      REAL LD                                                               CAL-0200
      WWW(BB,WTI)=(2.0*WTI)/(BB*BB)                                         CAL-0210
      XXYY(AA,BB,WTI,RI)=(-4.0*WTI*AA)/BB**3+(2.0*WTI*RI)/BB**2             CAL-0220
      ZZZZ(AA,BB,WTI,RI)=(6.0*WTI*AA**2)/(BB**4)-(4.0*WTI*RI*AA)/(BB**3)CAL-0230
C                                                                           CAL-0240
C                                                                           CAL-0250
      IF (IPER.EQ.LPP) GO TO 230                                            CAL-0260
      IF (IPER.NE.0) GO TO 260                                              CAL-0270
C                                                                           CAL-0280
C                                                                           CAL-0290
C     CALCULATE TU MATRIX (INVARIANT)                                       CAL-0300
C                                                                           CAL-0310
      DO 110 I=1,10                                                         CAL-0320
        DO 110 J=1,10                                                       CAL-0330
110     TU(I,J)=0.0                                                         CAL-0340
      DO 120 I=31,34                                                        CAL-0350
        TU(1,1)=TU(1,1)+WT(I)+WT(I+16)                                      CAL-0360
        TU(2,2)=TU(2,2)+WT(I+1)+WT(I+17)                                    CAL-0370
        TU(3,3)=TU(3,3)+WT(I+2)+WT(I+18)                                    CAL-0380
120     TU(4,4)=TU(4,4)+WT(I+3)+WT(I+19)                                    CAL-0390
      TU(5,1)=WT(31)+WT(47)                                                 CAL-0400
      TU(5,2)=WT(35)+WT(51)                                                 CAL-0410
      TU(5,3)=WT(39)+WT(55)                                                 CAL-0420
      TU(5,4)=WT(43)+WT(59)                                                 CAL-0430
      DO 130 I=31,59,4                                                      CAL-0440
130     TU(5,5)=TU(5,5)+WT(I)                                               CAL-0450
      TU(6,1)=WT(32)+WT(48)                                                 CAL-0460
      TU(6,2)=WT(36)+WT(52)                                                 CAL-0470
      TU(6,3)=WT(40)+WT(56)                                                 CAL-0480
      TU(6,4)=WT(44)+WT(60)                                                 CAL-0490
      DO 140 I=32,60,4                                                      CAL-0500
140     TU(6,6)=TU(6,6)+WT(I)                                               CAL-0510
      TU(7,1)=WT(33)+WT(49)                                                 CAL-0520
```

BEST COPY AVAILABLE


```
      TU(7,2)=WT(37)+WT(53)                                              CAL-0530
      TU(7,3)=WT(41)+WT(57)                                              CAL-0540
      TU(7,4)=WT(45)+WT(61)                                              CAL-0550
      DO 150 I=33,61,4                                                   CAL-0560
150     TU(7,7)=TU(7,7)+WT(I)                                            CAL-0570
      TU(8,1)=WT(34)+WT(50)                                              CAL-0580
      TU(8,2)=WT(38)+WT(54)                                              CAL-0590
      TU(8,3)=WT(42)+WT(58)                                              CAL-0600
      TU(8,4)=WT(46)+WT(62)                                              CAL-0610
      DO 160 I=34,62,4                                                   CAL-0620
160     TU(8,8)=TU(8,8)+WT(I)                                            CAL-0630
      DO 170 I=31,34                                                     CAL-0640
        TU(9,I-26)=WT(I)+WT(I+4)+WT(I+8)+WT(I+12)                        CAL-0650
170     TU(10,I-26)=WT(I+16)+WT(I+20)+WT(I+24)+WT(I+28)                  CAL-0660
      DO 180 I=31,34                                                     CAL-0670
        TU(9,1)=TU(9,1)+WT(I)                                            CAL-0680
        TU(9,2)=TU(9,2)+WT(I+1)                                          CAL-0690
        TU(9,3)=TU(9,3)+WT(I+2)                                          CAL-0700
        TU(9,4)=TU(9,4)+WT(I+3)                                          CAL-0710
        TU(10,1)=TU(10,1)+WT(I+16)                                       CAL-0720
        TU(10,2)=TU(10,2)+WT(I+17)                                       CAL-0730
        TU(10,3)=TU(10,3)+WT(I+18)                                       CAL-0740
        TU(10,4)=TU(10,4)+WT(I+19)                                       CAL-0750
180     CONTINUE                                                         CAL-0760
      DO 190 I=31,46                                                     CAL-0770
190     TU(9,9)=TU(9,9)+WT(I)                                            CAL-0780
      DO 200 I=47,62                                                     CAL-0790
200     TU(10,10)=TU(10,10)+WT(I)                                        CAL-0800
      DO 210 I=2,10                                                      CAL-0810
        M=I-1                                                            CAL-0820
        DO 210 J=1,M                                                     CAL-0830
210     TU(J,I)=TU(I,J)                                                  CAL-0840
      DO 220 I=1,10                                                      CAL-0850
        DO 220 J=1,10                                                    CAL-0860
220     TU(I,J)=2.0*TU(I,J)                                              CAL-0870
      RETURN                                                             CAL-0880
C                                                                        CAL-0890
C                                                                        CAL-0900
C     CALCULATE VXX2,VX2                                                 CAL-0910
C                                                                        CAL-0920
230   CONTINUE                                                           CAL-0930
      DO 240 I=1,2401                                                    CAL-0940
240     VXLIN(I)=0.0                                                     CAL-0950
      DO 250 I=1,NX                                                      CAL-0960
250     VX2(I)=0.0                                                       CAL-0970
      RETURN                                                             CAL-0980
C                                                                        CAL-0990
C                                                                        CAL-1000
260   CONTINUE                                                           CAL-1010
      DO 270 J=1,NX                                                      CAL-1020
        DO 270 I=1,NX                                                    CAL-1030
270     VXX(I,J)=0.0                                                     CAL-1040
```

BEST COPY AVAILABLE


```
      DO 280 J=1,NX                                                    CAL-1050
        DO 280 I=1,NU                                                  CAL-1060
280     VUX(I,J)=0.0                                                   CAL-1070
      DO 290 J=1,NU                                                    CAL-1080
        DO 290 I=1,NU                                                  CAL-1090
290     VUU(I,J)=0.0                                                   CAL-1100
      F1=X(1,IPER)+U(1,IPER)                                           CAL-1110
      F2=X(2,IPER)+U(2,IPER)                                           CAL-1120
      F3=X(3,IPER)+U(3,IPER)                                           CAL-1130
      F4=X(4,IPER)+U(4,IPER)                                           CAL-1140
      TF=F1+F2+F3+F4                                                   CAL-1150
      TSP=X(8,IPER)+X(12,IPER)                                         CAL-1160
      GRAD=X(45,IPER)+U(23,IPER)                                       CAL-1170
      LD=0.0                                                           CAL-1180
      DO 300 I=13,28                                                   CAL-1190
300     LD=LD+X(I,IPER)+U(I-6,IPER)                                    CAL-1200
      UD=0.0                                                           CAL-1210
      DO 310 I=29,44                                                   CAL-1220
310     UD=UD+X(I,IPER)+Z(I-28,IPER)                                   CAL-1230
      TS=LD+UD+GRAD                                                    CAL-1240
C                                                                      CAL-1250
C     TERMS 1,6,7 FACULTY-FACULTY PARTIALS CALCULATED IN Z167          CAL-1260
C                                                                      CAL-1270
      Z167=ZZZZ(TS,TF,WT(1),R(1))+2.0*WT(6)*SPACE(3)**2+2.0*WT(7)*SPACE(CAL-1280
     16)**2                                                            CAL-1290
      TEMP6E=2.0*WT(6)*SPACE(3)*SPACE(2)                               CAL-1300
      TEMP6F=2.0*WT(6)*SPACE(2)**2                                     CAL-1310
      TEMP7E=2.0*WT(7)*SPACE(6)*SPACE(5)                               CAL-1320
      TEMP7F=2.0*WT(7)*SPACE(5)**2                                     CAL-1330
      TEMP1W=WWWW(TF,WT(1))                                            CAL-1340
      TEMP1X=XXYY(TS,TF,WT(1),R(1))                                    CAL-1350
      TEMPX=TEMP1X+TEMP6E+TEMP7E                                       CAL-1360
      TEMPW=TEMP1W+TEMP6F+TEMP7F                                       CAL-1370
      DO 320 I=1,4                                                     CAL-1380
        DO 320 J=7,23                                                  CAL-1390
        VUX(J,I)=TEMPX                                                 CAL-1400
        VUU(J,I)=TEMPX                                                 CAL-1410
320     VUU(I,J)=TEMPX                                                 CAL-1420
      DO 330 I=1,4                                                     CAL-1430
        DO 330 J=13,45                                                 CAL-1440
        VUX(I,J)=TEMPX                                                 CAL-1450
        VXX(J,I)=TEMPX                                                 CAL-1460
330     VXX(I,J)=TEMPX                                                 CAL-1470
C                                                                      CAL-1480
C     TERMS 2,3,4  --DO UPPER HALF OF MATRIX                           CAL-1490
C                                                                      CAL-1500
      TEMP2W=WWWW(TF,R(2))                                             CAL-1510
      TEMP2X=XXYY(F1,TF,WT(2),R(2))                                    CAL-1520
      TEMP2Z=ZZZZ(F1,TF,WT(2),R(2))                                    CAL-1530
      TEMP3W=WWWW(TF,R(3))                                             CAL-1540
      TEMP3X=XXYY(F2,TF,WT(3),R(3))                                    CAL-1550
      TEMP3Z=ZZZZ(F2,TF,WT(3),R(3))                                    CAL-1560
```

BEST COPY AVAILABLE


```
      TEMP4W=WWWW(TF,R(4))                                          CAL-1570
      TEMP4X=XXYY(F3,TF,WT(4),R(4))                                 CAL-1580
      TEMP4Z=ZZZZ(F3,TF,WT(4),R(4))                                 CAL-1590
      ALL2=TEMP2W+2.0*TEMP2X                                        CAL-1600
      ALL3=TEMP3W+2.0*TEMP3X                                        CAL-1610
      ALL4=TEMP4W+2.0*TEMP4X                                        CAL-1620
      ALLZ=TEMP2Z+TEMP3Z+TEMP4Z+Z167                                CAL-1630
      VUU(1,1)=ALLZ+ALL2+TEMP3Z                                     CAL-1640
      VUU(1,2)=ALLZ+TEMP2X+TEMP3X                                   CAL-1650
      VUU(1,3)=ALLZ+TEMP2X+TEMP4X                                   CAL-1660
      VUU(1,4)=ALLZ+TEMP2X                                          CAL-1670
      VUU(2,2)=ALLZ+ALL3+TEMP2Z                                     CAL-1680
      VUU(2,3)=ALLZ+TEMP3X+TEMP4X                                   CAL-1690
      VUU(2,4)=ALLZ+TEMP3X                                          CAL-1700
      VUU(3,3)=ALLZ+ALL4+TEMP2Z                                     CAL-1710
      VUU(3,4)=ALLZ+TEMP4X                                          CAL-1720
      VUU(4,4)=ALLZ+TEMP2Z+TEMP3Z                                   CAL-1730
      DO 340 I=2,4                                                  CAL-1740
        M=I-1                                                       CAL-1750
        DO 340 J=1,M                                                CAL-1760
340     VUU(I,J)=VUU(J,I)                                           CAL-1770
      DO 350 I=1,4                                                  CAL-1780
        DO 350 J=1,4                                                CAL-1790
        VUX(I,J)=VUU(I,J)                                           CAL-1800
350     VXX(I,J)=VUU(I,J)                                           CAL-1810
C                                                                   CAL-1820
C     TERM 5   --VXX ONLY                                           CAL-1830
C                                                                   CAL-1840
      TEMP5W=WWWW(TSP,WT(5))                                        CAL-1850
      TEMP5X=XXYY(X(8,IPER),TSP,WT(5),R(5))                         CAL-1860
      TEMP5Z=ZZZZ(X(8,IPER),TSP,WT(5),R(5))                         CAL-1870
      VXX(8,8)=TEMP5W+TEMP5Z+2.0*TEMP5X                             CAL-1880
      VXX(12,12)=VXX(8,8)-TEMP5W                                    CAL-1890
      VXX(12,8)=VXX(12,12)-TEMP5X                                   CAL-1900
      VXX(8,12)=VXX(12,8)                                           CAL-1910
C                                                                   CAL-1920
C                                                                   CAL-1930
C     COMPLETE TERMS 6                                              CAL-1940
C                                                                   CAL-1950
      TEMP6C=-WT(6)*SPACE(3)                                        CAL-1960
      TEMP6B=-2.0*WT(6)*SPACE(2)                                    CAL-1970
      DO 360 I=1,4                                                  CAL-1980
360     VUX(I,8)=TEMP6C                                             CAL-1990
      DO 370 I=7,23                                                 CAL-2000
370     VUX(I,8)=TEMP6B                                             CAL-2010
      DO 380 I=1,4                                                  CAL-2020
        VXX(8,I)=TEMP6C                                             CAL-2030
380     VXX(I,8)=TEMP6C                                             CAL-2040
      DO 390 I=13,45                                                CAL-2050
        VXX(8,I)=TEMP6B                                             CAL-2060
390     VXX(I,8)=TEMP6B                                             CAL-2070
      VXX(8,8)=2.0*WT(6)                                            CAL-2080
```

BEST COPY AVAILABLE


```
C                                                                       CAL-2090
C     COMPLETE TERM 7                                                   CAL-2100
C                                                                       CAL-2110
      TEMP7C=-WT(7)*SPACE(6)                                            CAL-2120
      TEMP7B=-2.0*WT(7)*SPACE(5)                                        CAL-2130
      DO 400 I=1,4                                                      CAL-2140
400     VUX(I,12)=TEMP7C                                                CAL-2150
      DO 410 I=7,23                                                     CAL-2160
410     VUX(I,12)=TEMP7B                                                CAL-2170
      DO 420 I=1,4                                                      CAL-2180
        VXX(12,I)=TEMP7C                                                CAL-2190
420     VXX(I,12)=TEMP7C                                                CAL-2200
      DO 430 I=13,45                                                    CAL-2210
        VXX(12,I)=TEMP7B                                                CAL-2220
430     VXX(I,12)=TEMP7B                                                CAL-2230
      VXX(12,12)=2.0*WT(7)                                              CAL-2240
C                                                                       CAL-2250
C     TERM 8                                                            CAL-2260
C                                                                       CAL-2270
      VUU(38,38)=2.0*WT(8)                                              CAL-2280
C                                                                       CAL-2290
C     TERM 9                                                            CAL-2300
C                                                                       CAL-2310
      VXX(47,47)=2.0*WT(9)                                              CAL-2320
C                                                                       CAL-2330
C     TERM 10                                                           CAL-2340
C                                                                       CAL-2350
      TEMP=2.0*WT(10)                                                   CAL-2360
      VXX(48,48)=TEMP                                                   CAL-2370
      VXX(48,49)=TEMP                                                   CAL-2380
      VXX(49,48)=TEMP                                                   CAL-2390
      VXX(49,49)=TEMP                                                   CAL-2400
C                                                                       CAL-2410
C     TERMS 11 12--TEMPW COMES FROM TERMS 1,6,7                         CAL-2420
C                                                                       CAL-2430
      TEM11W=WWWW(TS,WT(12))                                            CAL-2440
      TEM11X=XXYY(LD,TS,WT(12),R(12))                                   CAL-2450
      TEM11Z=ZZZZ(LD,TS,WT(12),R(12))                                   CAL-2460
      TEM12W=WWWW(TS,WT(13))                                            CAL-2470
      TEM12X=XXYY(UD,TS,WT(13),R(13))                                   CAL-2480
      TEM12Z=ZZZZ(UD,TS,WT(13),R(13))                                   CAL-2490
      ALL11=TEM11W+TEM11Z+2.0*TEM11X                                    CAL-2500
      ALL12=TEM12W+TEM12Z+2.0*TEM12X                                    CAL-2510
      TEMP=ALL11+TEM12Z+TEMPW                                           CAL-2520
      DO 440 I=7,22                                                     CAL-2530
        DO 440 J=7,22                                                   CAL-2540
440     VUU(I,J)=TEMP                                                   CAL-2550
      TEMP=TEM11X+TEM11Z+TEM12Z+TEMPW                                   CAL-2560
      DO 450 I=7,22                                                     CAL-2570
        VUU(23,I)=TEMP                                                  CAL-2580
450     VUU(I,23)=TEMP                                                  CAL-2590
      VUU(23,23)=TEM11Z+TEM12Z+TEMPW                                    CAL-2600
```

BEST COPY AVAILABLE


```
      TEMP=ALL11+TEM12Z+TEMPW                                          CAL-2610
      DO 460 I=13,28                                                   CAL-2620
        DO 460 J=13,28                                                 CAL-2630
460     VXX(I,J)=TEMP                                                  CAL-2640
      DO 470 I=7,22                                                    CAL-2650
        DO 470 J=13,28                                                 CAL-2660
470     VUX(I,J)=TEMP                                                  CAL-2670
      TEMP=ALL12+TEM11Z+TEMPW                                          CAL-2680
      DO 480 I=29,44                                                   CAL-2690
        DO 480 J=29,44                                                 CAL-2700
480     VXX(I,J)=TEMP                                                  CAL-2710
      TEMP=TEM11X+TEM11Z+TEM12X+TEM12Z+TEMPW                           CAL-2720
      DO 490 I=29,44                                                   CAL-2730
        DO 490 J=13,28                                                 CAL-2740
        VXX(I,J)=TEMP                                                  CAL-2750
490     VXX(J,I)=TEMP                                                  CAL-2760
      DO 500 I=7,22                                                    CAL-2770
        DO 500 J=29,44                                                 CAL-2780
500     VUX(I,J)=TEMP                                                  CAL-2790
      TEMP=TEM11X+TEM11Z+TEM12Z+TEMPW                                  CAL-2800
      DO 510 I=13,28                                                   CAL-2810
        VXX(45,I)=TEMP                                                 CAL-2820
510     VXX(I,45)=TEMP                                                 CAL-2830
      DO 520 J=13,44                                                   CAL-2840
520     VUX(23,J)=TEMP                                                 CAL-2850
      DO 530 I=7,22                                                    CAL-2860
530     VUX(I,45)=TEMP                                                 CAL-2870
      TEMP=TEM12X+TEM12Z+TEM11Z+TEMPW                                  CAL-2880
      DO 540 I=29,44                                                   CAL-2890
        VXX(45,I)=TEMP                                                 CAL-2900
540     VXX(I,45)=TEMP                                                 CAL-2910
      VXX(45,45)=TEM11Z+TEM12Z                                         CAL-2920
      VUX(23,45)=VXX(45,45)                                            CAL-2930
C                                                                      CAL-2940
C     TERM 13                                                          CAL-2950
C                                                                      CAL-2960
      DO 550 I=7,22                                                    CAL-2970
550     VUU(I,I)=2.0*WT(I+7)+VUU(1,I)                                  CAL-2980
C                                                                      CAL-2990
C     TERM 14 --CODE TO INSERT MATRIX TU(INVARIANT)                    CAL-3000
C                                                                      CAL-3010
      DO 560 I=1,10                                                    CAL-3020
        DO 560 J=1,10                                                  CAL-3030
560     VUU(I+26,J+26)=TU(I,J)                                         CAL-3040
      VUU(37,37)=2.0*WT(63)                                            CAL-3050
C                                                                      CAL-3060
C     TERM 15                                                          CAL-3070
C                                                                      CAL-3080
      VUU(5,5)=WWWW(X(8,IPER),WT(64))                                  CAL-3090
      VXX(8,8)=ZZZZ(U(5,IPER),X(8,IPER),WT(64),R(64))                  CAL-3100
      VUX(5,8)=XXYY(U(5,IPER),X(8,IPER),WT(64),R(64))                  CAL-3110
C                                                                      CAL-3120
```

BEST COPY AVAILABLE


```
C       TERM 16                                                          CAL-3130
C                                                                        CAL-3140
        VUU(6,6)=WWWW(X(12,IPER),WT(65))                                 CAL-3150
        VXX(12,12)=ZZZZ(U(6,IPER),X(12,IPER),WT(65),R(65))               CAL-3160
        VUX(6,12)=XXYY(U(6,IPER),X(12,IPER),WT(65),R(65))                CAL-3170
C                                                                        CAL-3180
C       TERM 17                                                          CAL-3190
C                                                                        CAL-3200
        VUU(24,24)=2.0*WT(66)                                            CAL-3210
C                                                                        CAL-3220
C       TERM 18                                                          CAL-3230
C                                                                        CAL-3240
        VUU(25,25)=2.0*WT(67)                                            CAL-3250
C                                                                        CAL-3260
C       TERM 19                                                          CAL-3270
C                                                                        CAL-3280
        VUU(26,26)=2.0*WT(68)                                            CAL-3290
C                                                                        CAL-3300
C                                                                        CAL-3310
C       THE FOLLOWING CODE ADDS (VX2,FUU) AND ( VX2,FUX) TO VUU          CAL-3320
C       AND VUX, RESPECTIVELY.                                           CAL-3330
        TEMP=VX2(49)                                                     CAL-3340
        N=7                                                              CAL-3350
        DO 580 I=27,30                                                   CAL-3360
          NLAST=N+3                                                      CAL-3370
          DO 570 J=N,NLAST                                               CAL-3380
            VUX(I,J+6)=VUX(I,J+6)+TEMP                                   CAL-3390
            VUX(I,J+22)=VUX(I,J+22)+TEMP                                 CAL-3400
            VUU(I,J)=VUU(I,J)+TEMP                                       CAL-3410
570         VUU(J,I)=VUU(J,I)+TEMP                                       CAL-3420
580       N=N+4                                                          CAL-3430
        N=7                                                              CAL-3440
        DO 600 I=31,34                                                   CAL-3450
          NLAST=N+12                                                     CAL-3460
          DO 590 J=N,NLAST,4                                             CAL-3470
            VUX(I,J+6)=VUX(I,J+6)+TEMP                                   CAL-3480
            VUX(I,J+22)=VUX(I,J+22)+TEMP                                 CAL-3490
            VUU(I,J)=VUU(I,J)+TEMP                                       CAL-3500
590         VUU(J,I)=VUU(J,I)+TEMP                                       CAL-3510
          N=N+1                                                          CAL-3520
600       CONTINUE                                                       CAL-3530
        DO 610 J=7,22                                                    CAL-3540
          VUX(35,J+6)=VUX(35,J+6)+TEMP                                   CAL-3550
          VUU(35,J)=VUU(35,J)+TEMP                                       CAL-3560
          VUU(J,35)=VUU(J,35)+TEMP                                       CAL-3570
          VUX(38,J+6)=VUX(38,J+6)+TEMP                                   CAL-3580
          VUU(38,J)=VUU(38,J)+TEMP                                       CAL-3590
610       VUU(J,38)=VUU(J,38)+TEMP                                       CAL-3600
        VUU(37,23)=VUU(37,23)+TEMP                                       CAL-3610
        VUU(23,37)=VUU(23,37)+TEMP                                       CAL-3620
        VUU(38,23)=VUU(38,23)+TEMP                                       CAL-3630
        VUU(23,38)=VUU(23,38)+TEMP                                       CAL-3640
```

BEST COPY AVAILABLE


```
      DO 620 J=29,44                                                    CAL-3650
        VUX(35,J)=VUX(35,J)+TEMP                                        CAL-3660
620     VUX(38,J)=VUX(35,J)+TEMP                                        CAL-3670
      VUX(37,45)=VUX(37,45)+TEMP                                        CAL-3680
      VUX(38,45)=VUX(38,45)+TEMP                                        CAL-3690
C                                                                       CAL-3700
C     CALCULATE VX,VU                                                   CAL-3710
C                                                                       CAL-3720
C                                                                       CAL-3730
C     GLOBAL TERMS FOR CALCULATION OF VU,VX                             CAL-3740
C                                                                       CAL-3750
      TEMP=2.0*WT(5)*(X(8,IPER)/TSP-R(5))                               CAL-3760
      TERMYY=TEMP*(TSP-X(8,IPER))/TSP**2                                CAL-3770
      TERMXX=TEMP*X(8,IPER)/TSP**2                                      CAL-3780
      TEMP=-2.0*WT(13)*(UC/TS-R(13))                                    CAL-3790
      TERMA=TEMP*UC/TS**2                                               CAL-3800
      TERMAA=TEMP*(TS-UC)/TS**2                                         CAL-3810
      TEMP=-2.0*WT(12)*(LC/TS-R(12))                                    CAL-3820
      TERMB=TEMP*(TS-LC)/TS**2                                          CAL-3830
      TERMBB=TEMP*UC/TS**2                                              CAL-3840
      TEMP=2.0*WT(1)*(TS/TF-R(1))                                       CAL-3850
      TERMC=TEMP/TF                                                     CAL-3860
      TERMD=-TEMP*(TS/TF**2)                                            CAL-3870
      TEMP=2.0*WT(2)*(F1/TF-R(2))                                       CAL-3880
      TERME=-TEMP*(F1/TF**2)                                            CAL-3890
      TERMF=TEMP*(TF-F1)/TF**2                                          CAL-3900
      TEMP=2.0*WT(3)*(F2/TF-R(3))                                       CAL-3910
      TERMG=-TEMP*F2/TF**2                                              CAL-3920
      TERMH=TEMP*(TF-F2)/TF**2                                          CAL-3930
      TEMP=2.0*WT(4)*(F3/TF-R(4))                                       CAL-3940
      TERMI=-TEMP*F3/TF**2                                              CAL-3950
      TERMJ=TEMP*(TF-F3)/TF**2                                          CAL-3960
      C123=SPACE(1)+SPACE(2)*TS+SPACE(3)*TF-X(8,IPER)                   CAL-3970
      C456=SPACE(4)+SPACE(5)*TS+SPACE(6)*TF-X(12,IPER)                  CAL-3980
      TERMK=2.0*WT(6)*C123*SPACE(3)                                     CAL-3990
      TERML=2.0*WT(6)*C123*SPACE(2)                                     CAL-4000
      TERMM=2.0*WT(7)*C456*SPACE(6)                                     CAL-4010
      TERMN=2.0*WT(7)*C456*SPACE(5)                                     CAL-4020
C                                                                       CAL-4030
C     FILL VECTOR SU                                                    CAL-4040
C                                                                       CAL-4050
      L=30                                                              CAL-4060
      M=0                                                               CAL-4070
      N=15                                                              CAL-4080
      TEMP1=U(35,IPER)                                                  CAL-4090
      TEMP2=U(36,IPER)                                                  CAL-4100
      DO 630 I=27,30                                                    CAL-4110
        TEMP3=U(I,IPER)+TEMP1                                           CAL-4120
        TEMP4=U(I,IPER)+TEMP2                                           CAL-4130
        DO 630 J=31,34                                                  CAL-4140
        M=M+1                                                           CAL-4150
        N=N+1                                                           CAL-4160
```

BEST COPY AVAILABLE


```
        L=L+1                                                          CAL-4170
        SU(M)=(TEMP3+U(J,IPER)-R(L))*WT(L)*2.0                         CAL-4180
        SU(N)=(TEMP4+U(J,IPER)-R(L+16))*WT(L+16)*2.0                   CAL-4190
630     CONTINUE                                                       CAL-4200
C                                                                      CAL-4210
C     CODE TO CALCULATE VU,VX                                          CAL-4220
C                                                                      CAL-4230
C                                                                      CAL-4240
      TEMP=TERMD+TERMK+TERMM                                           CAL-4250
      VX(1)=TERMF+TERMG+TERMI+TEMP                                     CAL-4260
      VU(1)=VX(1)                                                      CAL-4270
      VX(2)=TERME+TERMH+TERMI+TEMP                                     CAL-4280
      VU(2)=VX(2)                                                      CAL-4290
      VX(3)=TERME+TERMG+TERMJ+TEMP                                     CAL-4300
      VU(3)=VX(3)                                                      CAL-4310
      VX(4)=TERME+TERMG+TERMI+TEMP                                     CAL-4320
      VU(4)=VX(4)                                                      CAL-4330
      VX(5)=0.0                                                        CAL-4340
      VX(6)=0.0                                                        CAL-4350
      VX(7)=0.0                                                        CAL-4360
      VX(9)=0.0                                                        CAL-4370
      VX(10)=0.0                                                       CAL-4380
      VX(11)=0.0                                                       CAL-4390
      TEMP1=2.0*WT(64)*(U(5,IPER)/X(8,IPER)-R(64))                     CAL-4400
      TEMP2=2.0*WT(65)*(U(6,IPER)/X(12,IPER)-R(65))                    CAL-4410
      VX(8)=-TEMP1*(U(5,IPER)/X(8,IPER)**2)+TERMYY                     CAL-4420
      VX(12)=-TEMP2*(U(6,IPER)/X(12,IPER)**2)+TERMXX                   CAL-4430
      VU(5)=TEMP1/X(8,IPER)                                            CAL-4440
      VU(6)=TEMP2/X(12,IPER)                                           CAL-4450
      TEMP1=TERMC+TERML+TERMN                                          CAL-4460
      TEMP2=TEMP1+TERMA+TERMB                                          CAL-4470
      DO 640 I=13,28                                                   CAL-4480
        VX(I)=TEMP2                                                    CAL-4490
640     VU(I-6)=TEMP2+2.0*WT(I+1)*(U(I-6,IPER)-R(I+1))                 CAL-4500
      TEMP2=TEMP1+TERMBB+TERMAA                                        CAL-4510
      DO 650 I=29,44                                                   CAL-4520
650     VX(I)=TEMP2                                                    CAL-4530
      VX(45)=TEMP1+TERMBB+TERMA                                        CAL-4540
      VU(23)=TEMP1+TERMA                                               CAL-4550
      VX(46)=0.0                                                       CAL-4560
      VX(47)=2.0*WT(9)*(X(47,IPER)-R(9))                               CAL-4570
      VX(48)=2.0*WT(10)*(X(48,IPER)+X(49,IPER)-R(10))                  CAL-4580
      VX(49)=VX(48)                                                    CAL-4590
      VU(24)=2.0*WT(66)                                                CAL-4600
      VU(25)=2.0*WT(67)                                                CAL-4610
      VU(26)=2.0*WT(68)                                                CAL-4620
      VU(37)=2.0*WT(63)*(U(37,IPER)-R(63))                             CAL-4630
      VU(38)=2.0*WT(8)*(U(38,IPER)-R(8))                               CAL-4640
C                                                                      CAL-4650
      DO 660 I=27,36                                                   CAL-4660
660     VU(I)=0.0                                                      CAL-4670
      DO 670 I=1,4                                                     CAL-4680
```

BEST COPY AVAILABLE


```
670       VU(27)=VU(27)+SU(I)+SU(I+16)                          CAL-4690
          DO 680 I=5,8                                          CAL-4700
680       VU(28)=VU(28)+SU(I)+SU(I+16)                          CAL-4710
          DO 690 I=9,12                                         CAL-4720
690       VU(29)=VU(29)+SU(I)+SU(I+16)                          CAL-4730
          DO 700 I=13,16                                        CAL-4740
700       VU(30)=VU(30)+SU(I)+SU(I+16)                          CAL-4750
          DO 710 I=1,29,4                                       CAL-4760
710       VU(31)=VU(31)+SU(I)                                   CAL-4770
          DO 720 I=2,30,4                                       CAL-4780
720       VU(32)=VU(32)+SU(I)                                   CAL-4790
          DO 730 I=3,31,4                                       CAL-4800
730       VU(33)=VU(33)+SU(I)                                   CAL-4810
          DO 740 I=4,32,4                                       CAL-4820
740       VU(34)=VU(34)+SU(I)                                   CAL-4830
          DO 750 I=1,16                                         CAL-4840
750       VU(35)=VU(35)+SU(I)                                   CAL-4850
          DO 760 I=17,32                                        CAL-4860
760       VU(36)=VU(36)+SU(I)                                   CAL-4870
C                                                               CAL-4880
          RETURN                                                CAL-4890
          END                                                   CAL-4900
```

BEST COPY AVAILABLE


```
      SUBROUTINE MDUMP(NAME,ATRIX,ISIZE,IMAX,JMAX)                    MDU-0010
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                         MDU-0020
      DIMENSION ATRIX(ISIZE,JMAX)                                     MDU-0030
      WRITE(IOUT,101)NAME                                             MDU-0040
      DO  10  I=1,IMAX                                                MDU-0050
      WRITE(IOUT,100)I,(ATRIX(I,J),J=1,JMAX)                          MDU-0060
10    CONTINUE                                                        MDU-0070
      RETURN                                                          MDU-0080
100   FORMAT(/1X,I3,10E11.0/(4X,10E11.0))                             MDU-0090
101   FORMAT(16H1DUMP OF MATRIX  ,A4///)                              MDU-0100
      END                                                             MDU-0110
```

BEST COPY AVAILABLE


```
      SUBROUTINE VDUMP(NAME,VECTOR,IMAX)                          VDU-0010
      COMMON/FILES/IIN,ICONI,ICONO,IOUT,IDUMP                     VDU-0020
      DIMENSION VECTOR(100)                                       VDU-0030
      WRITE(IOUT,101)NAME                                         VDU-0040
      WRITE(IOUT,100)(VECTOR(I),I=1,IMAX)                         VDU-0050
      RETURN                                                      VDU-0060
101   FORMAT(//////16H DUMP OF VECTOR  ,A4///)                    VDU-0070
100   FORMAT(4X,10E11.0)                                          VDU-0080
      END                                                         VDU-0090
```

BEST COPY AVAILABLE


```
      FUNCTION LINEQU(C,INDEX,IROWD,IROW,ICOL,TOL)                      LIN-0010
C                                                                       LIN-0020
C  ON ENTRY...                                                          LIN-0030
C        C IS THE ADDRESS OF THE MATRIX TO BE INVERTED                  LIN-0040
C        INDEX IS THE ADDRESS OF A VECTOR FOR SCRATCH USE IN PIVOTING   LIN-0050
C        IROWD IS THE ROW DIMENSION OF C IN THE CALLING PROGRAM         LIN-0060
C        IROW IS THE NUMBER OF ROWS OF C                                LIN-0070
C        ICOL IS THE NUMBER OF COLUMNS OF THE AUGMENTED MATRIX          LIN-0080
C        TOL IS A TOLERANCE VALUE TO BE USED IN CHECKING FOR            LIN-0090
C                 SINGULAR MATRICES                                     LIN-0100
C                                                                       LIN-0110
C   N.B.   THE SPACE C IS USED FOR SCRATCH DURING THE COMPUTATION,      LIN-0120
C  AND MUST HAVE SIZE OF AT LEAST IROWD*ICOL.                           LIN-0130
C                                                                       LIN-0140
C  COMPUTATION METHOD IS STANDARD ELIMINATION USING ROW PIVOTING.       LIN-0150
C  THE INVERTED MATRIX (IF IT EXISTS) RESIDES IN C AT TERMINATION       LIN-0160
C                                                                       LIN-0170
C        LINEQU=0 IS NORMAL INVERSION TERMINATION                       LIN-0180
C        LINEQU=1 IS ERROR CONDITION(MATRIX IS SINGULAR)                LIN-0190
C                                                                       LIN-0200
      DIMENSION C(IROWD,ICOL)                                           LIN-0210
      DIMENSION INDEX(IROWD)                                            LIN-0220
C                                                                       LIN-0230
C  SET UP IDENTITY PART OF AUGMENTED MATRIX AND INDEX                   LIN-0240
      IL=IROW+1                                                         LIN-0250
      DO  10  I=1,IROW                                                  LIN-0260
      DO  10  J=IL,ICOL                                                 LIN-0270
10    C(I,J)=0.0                                                        LIN-0280
      DO  20  I=1,IROW                                                  LIN-0290
      INDEX(I)=I                                                        LIN-0300
      IR=I+IROW                                                         LIN-0310
20    C(I,IR)=1.0                                                       LIN-0320
C                                                                       LIN-0330
C  OUTER LOOP ESTABLISHES THE INDEX OF THE CURRENT PIVOT ELEMENT        LIN-0340
      DO  30  I=1,IROW                                                  LIN-0350
      IMAX=0                                                            LIN-0360
      VMAX=0.0                                                          LIN-0370
C                                                                       LIN-0380
C  FOLLOWING LOOP SEARCHES FOR LARGEST PIVOT ELEMENT                    LIN-0390
      DO  40  L=I,IROW                                                  LIN-0400
      LX=INDEX(L)                                                       LIN-0410
      TEMP=ABS(C(LX,I))                                                 LIN-0420
      IF(TEMP .LT. VMAX) GO TO 40                                       LIN-0430
      VMAX=TEMP                                                         LIN-0440
      IMAX=L                                                            LIN-0450
40    CONTINUE                                                          LIN-0460
C                                                                       LIN-0470
C  INTERCHANGE ROWS TO GET PIVOT ROW NEXT                               LIN-0480
      IF(IMAX .EQ. 0) GO TO 950                                         LIN-0490
      IF(VMAX .LT. TOL) GO TO 950                                       LIN-0500
      IX=INDEX(IMAX)                                                    LIN-0510
      IF(IMAX .EQ. I) GO TO 50                                          LIN-0520
```

BEST COPY AVAILABLE


```
      INDEX(IMAX)=INDEX(I)                                              LIN-0530
      INDEX(I)=IX                                                       LIN-0540
50    CONTINUE                                                          LIN-0550
C                                                                       LIN-0560
C  DIVIDE PIVOT ROW BY PIVOT ELEMENT                                    LIN-0570
      P=1.0/C(IX,I)                                                     LIN-0580
      DO  60  J=I,ICOL                                                  LIN-0590
60    C(IX,J)=C(IX,J)*P                                                 LIN-0600
C                                                                       LIN-0610
C  REMAINING TWO LOOPS SUBTRACT MULTIPLES OF THE PIVOT ROW              LIN-0620
C  COEFFICIENTS FROM ROW COEFFICIENTS IN ALL OTHER ROWS.                LIN-0630
      DO  70  L=1,IROW                                                  LIN-0640
      LX=INDEX(L)                                                       LIN-0650
      IF(L .EQ. I) GO TO 70                                             LIN-0660
      TEMP=C(LX,I)                                                      LIN-0670
      DO  75   J=1,ICOL                                                 LIN-0680
75    C(LX,J)=C(LX,J)-TEMP*C(IX,J)                                      LIN-0690
70    CONTINUE                                                          LIN-0700
30    CONTINUE                                                          LIN-0710
C                                                                       LIN-0720
C  INVERSION COMPLETED--REMAINING CODE TO UNWIND ROWS AND PLACE         LIN-0730
C  INVERSE IN C.                                                        LIN-0740
      DO   80  I=1,IROW                                                 LIN-0750
      IX=INDEX(I)                                                       LIN-0760
      DO  80  J=1,IROW                                                  LIN-0770
      JR=J+IROW                                                         LIN-0780
80    C(I,J)=C(IX,JR)                                                   LIN-0790
C                                                                       LIN-0800
C  SUCCESS EXIT                                                         LIN-0810
900   LINEQU=0                                                          LIN-0820
      RETURN                                                            LIN-0830
C                                                                       LIN-0840
C  ERROR EXIT                                                           LIN-0850
950   LINEQU=1                                                          LIN-0860
      RETURN                                                            LIN-0870
      END                                                               LIN-0880
```

BEST COPY AVAILABLE


```
      PROGRAM MODIFY (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,TAPE2)MOD-0010
C                                                                       MOD-0020
C                                                                       MOD-0030
C           MODIFY -- VERSION 3 (CDC 6000)  3/73                        MOD-0040
C                                                                       MOD-0050
C     --PREPARED FOR UNITED STATES OFFICE OF EDUCATION BY THE           MOD-0060
C     --NATIONAL CENTER FOR HIGHER EDUCATION MANAGEMENT SYSTEMS         MOD-0070
C     --AT WICHE UNDER USOE CONTRACT-                                   MOD-0080
C                                                                       MOD-0090
C                       OEC-0-72-3575                                   MOD-0100
C                                                                       MOD-0110
C                                                                       MOD-0120
C     THIS DATA BASE MODIFICATION PROGRAM IS PART OF THE NATIONAL       MOD-0130
C     PLANNING MODEL PROJECT AT NCHEMS.                                 MOD-0140
C                                                                       MOD-0150
C                                                                       MOD-0160
C        ***NOTE***      NO WARRANTY, EITHER EXPRESS OR IMPLIED,        MOD-0170
C                        IS MADE FOR THIS COLLECTION OF PROGRAMS        MOD-0180
C                        OR FOR ITS ACCOMPANYING TEST DATA.             MOD-0190
C                        ALL CURRENT SOFTWARE FOR THIS PROJECT IS       MOD-0200
C                        RELEASED AS NCHEMS TYPE II SOFTWARE,           MOD-0210
C                        WITH NO PROVISION MADE FOR PROGRAM ACCURACY    MOD-0220
C                        OR MAINTENANCE.                                MOD-0230
C                                                                       MOD-0240
C                                                                       MOD-0250
      LOGICAL OLDFIL,PRT                                                MOD-0260
      LOGICAL PUNCH                                                     MOD-0270
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIMOD-0280
     1N,IPUNCH,PUNCH                                                    MOD-0290
      DATA ICHARY/1HY/                                                  MOD-0300
      DATA ICHARI,ICHARC/1HI,1HC/                                       MOD-0310
      DATA IRTPS/2,5*0,60*0,4*4/                                        MOD-0320
      ICONI=5                                                           MOD-0330
      ICONO=6                                                           MOD-0340
      IIN=1                                                             MOD-0350
      IPUNCH=2                                                          MOD-0360
      DO 105 I=7,62,5                                                   MOD-0370
        IRTPS(I)=1                                                      MOD-0380
        IRTPS(I+1)=5                                                    MOD-0390
        IRTPS(I+2)=3                                                    MOD-0400
        IRTPS(I+3)=6                                                    MOD-0410
105     IRTPS(I+4)=6                                                    MOD-0420
      CALL DATA (1)                                                     MOD-0430
      WRITE (ICONO,165)                                                 MOD-0440
      PRT=.FALSE.                                                       MOD-0450
      WRITE (ICONO,155)                                                 MOD-0460
      READ (ICONI,185) INVAL                                            MOD-0470
      IF (INVAL.EQ.ICHARI) PRT=.TRUE.                                   MOD-0480
      PUNCH=.FALSE.                                                     MOD-0490
      WRITE (ICONO,160)                                                 MOD-0500
      READ , INVAL                                                      MOD-0510
      IF (INVAL.EQ.ICHARY) PUNCH=.TRUE.                                 MOD-0520
```

BEST COPY AVAILABLE


```
      OLDFIL=.TRUE.                                                    MOD-0530
      WRITE (ICONO,175)                                                MOD-0540
      READ (ICONI,185) INVAL                                           MOD-0550
      IF (INVAL.EQ.ICHARC) OLDFIL=.FALSE.                              MOD-0560
110   WRITE (ICONO,190)                                                MOD-0570
C     NON-ANSI UNFORMATTED READ                                        MOD-0580
      READ , KEY                                                       MOD-0590
      IF (PUNCH) WRITE (IPUNCH,180) KEY                                MOD-0600
      IF (KEY.EQ.0) GO TO 150                                          MOD-0610
      IF (KEY.GT.70.OR.KEY.LE.0) GO TO 145                             MOD-0620
      K=IRTPS(KEY)                                                     MOD-0630
      IF (K.NE.0) GO TO (115,120,125,130,135,140), K                   MOD-0640
      GO TO 145                                                        MOD-0650
115   CALL MOD1                                                        MOD-0660
      GO TO 110                                                        MOD-0670
120   CALL MOD2                                                        MOD-0680
      GO TO 110                                                        MOD-0690
125   CALL MOD3                                                        MOD-0700
      GO TO 110                                                        MOD-0710
130   CALL MOD4                                                        MOD-0720
      GO TO 110                                                        MOD-0730
135   CALL MOD5                                                        MOD-0740
      GO TO 110                                                        MOD-0750
140   CALL MOD6                                                        MOD-0760
      GO TO 110                                                        MOD-0770
145   WRITE (ICONO,170)                                                MOD-0780
      GO TO 110                                                        MOD-0790
150   CALL DATA (2)                                                    MOD-0800
      WRITE (ICONO,195)                                                MOD-0810
      STOP                                                             MOD-0820
C                                                                      MOD-0830
155   FORMAT (33H BATCH OR INTERACTIVE JOB(B OR I))                    MOD-0840
160   FORMAT (31H CHANGES TO BE PUNCHED (Y OR N))                      MOD-0850
165   FORMAT (10X,25HMODIFY--VERSION 3--JAN 73//)                      MOD-0860
170   FORMAT (28H BAD RECORD NUMBER,TRY AGAIN)                         MOD-0870
175   FORMAT (45H CREATE NEW FILE OR MODIFY AN OLD ONE(C OR M))        MOD-0880
180   FORMAT (I3)                                                      MOD-0890
185   FORMAT (A1)                                                      MOD-0900
190   FORMAT (22H TYPE IN RECORD NUMBER)                               MOD-0910
195   FORMAT (53H EDITING COMPLETE--OUTPUT ON UNIT 1, PUNCH ON UNIT 2.) MOD-0920
      END                                                              MOD-0930
```

BEST COPY AVAILABLE


```
      SUBROUTINE MOD1                                                   MD1-0010
      LOGICAL OLDFIL,PRT                                                MD1-0020
      LOGICAL PUNCH                                                     MD1-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIMD1-0040
     1N,IPUNCH,PUNCH                                                    MD1-0050
      DIMENSION XF(12)                                                  MD1-0060
      COMMON INDIF(150),INDJF(150),VALF(150),INDIG(150),INDJG(150),VALG(MD1-0070
     1150),NUMF,NUMG,WT(70),R(70),SPACE(6),H(10),TRANSF,NF,NG           MD1-0080
      COMMON EXTRA(3100)                                                MD1-0090
      DIMENSION NAMES(6)                                                MD1-0100
      DATA XF/2*.TRUE.,10*.FALSE./                                      MD1-0110
      DATA ICHARO/1HO/                                                  MD1-0120
      DATA NAMES/1HF,1HG,1HH,1Hw,1HR,1HS/                               MD1-0130
      DATA NX,NU,NZ,NPP,MAXK,MAXR/49,38,63,5,2*69/                      MD1-0140
      DATA NH/2/                                                        MD1-0150
      IF (PRT) WRITE (ICONO,150)                                        MD1-0160
      NUMF=0                                                            MD1-0170
      NUMG=0                                                            MD1-0180
      IF (OLDFIL) CALL DATA (13)                                        MD1-0190
105   IF (PRT) WRITE (ICONO,160)                                        MD1-0200
      READ (ICONI,155) INVAL                                            MD1-0210
      IF (PUNCH) WRITE (IPUNCH,155) INVAL                               MD1-0220
      IF (INVAL.EQ.ICHARO) GO TO 145                                    MD1-0230
      DO 110 I=1,6                                                      MD1-0240
        IF (INVAL.EQ.NAMES(I)) GO TO (115,120,125,130,135,140), I       MD1-0250
110     CONTINUE                                                        MD1-0260
      GO TO 105                                                         MD1-0270
115   CALL SPEC (INDIF,INDJF,VALF,NX,NX,NUMF,EXTRA)                     MD1-0280
      GO TO 105                                                         MD1-0290
120   CALL SPEC (INDIG,INDJG,VALG,NX,NU,NUMG,EXTRA)                     MD1-0300
      GO TO 105                                                         MD1-0310
125   CALL S (H,NH)                                                     MD1-0320
      GO TO 105                                                         MD1-0330
130   CALL S (WT,MAXK)                                                  MD1-0340
      GO TO 105                                                         MD1-0350
135   CALL S (R,MAXR)                                                   MD1-0360
      GO TO 105                                                         MD1-0370
140   CALL S (SPACE,6)                                                  MD1-0380
      GO TO 105                                                         MD1-0390
145   IWH=(KEY-2)/5                                                     MD1-0400
      TRANSF=XF(IWH)                                                    MD1-0410
      CALL DATA (14)                                                    MD1-0420
      RETURN                                                            MD1-0430
C                                                                       MD1-0440
150   FORMAT (39H MOD1-TRANSITION MATRICES AND CONSTANTS)               MD1-0450
155   FORMAT (A1)                                                       MD1-0460
160   FORMAT (30H TYPE IN MATRIX OR VECTOR NAME)                        MD1-0470
      END                                                               MD1-0480
```

BEST COPY AVAILABLE


```
      SUBROUTINE MOD2                                                    MD2-0010
      LOGICAL PUNCH                                                      MD2-0020
      LOGICAL OLDFIL,PRT                                                 MD2-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIMD2-0040
     1N,IPUNCH,PUNCH                                                     MD2-0050
      REAL NHLAST,NHS                                                    MD2-0060
      DIMENSION AN(11)                                                   MD2-0070
      EQUIVALENCE (AN,NHLAST)                                            MD2-0080
      COMMON NHLAST,NHS(10),PCT(16),ACH(16),BETA1(4),BETA2(4),Y(16)      MD2-0090
      DIMENSION NAMES(6)                                                 MD2-0100
      DATA NAMES/1HN,1HP,1HA,1HB,1HC,1HY/                                MD2-0110
      DATA NUM/6/                                                        MD2-0120
      DATA ICHARO/1HO/                                                   MD2-0130
      IF (PRT) WRITE (ICONO,150)                                         MD2-0140
      IF (OLDFIL) CALL DATA (5)                                          MD2-0150
105   IF (PRT) WRITE (ICONO,160)                                         MD2-0160
      READ (ICONI,155) INVAL                                             MD2-0170
      IF (PUNCH) WRITE (IPUNCH,155) INVAL                                MD2-0180
      IF (INVAL.EQ.ICHARO) GO TO 145                                     MD2-0190
      DO 110 I=1,NUM                                                     MD2-0200
        IF (INVAL.EQ.NAMES(I)) GO TO (115,120,125,130,135,140), I        MD2-0210
110     CONTINUE                                                         MD2-0220
      GO TO 105                                                          MD2-0230
115   CONTINUE                                                           MD2-0240
      CALL S (AN,11)                                                     MD2-0250
      GO TO 105                                                          MD2-0260
120   CALL S (PCT,16)                                                    MD2-0270
      GO TO 105                                                          MD2-0280
125   CALL S (ACH,16)                                                    MD2-0290
      GO TO 105                                                          MD2-0300
130   CALL S (BETA1,4)                                                   MD2-0310
      GO TO 105                                                          MD2-0320
135   CALL S (BETA2,4)                                                   MD2-0330
      GO TO 105                                                          MD2-0340
140   CALL S (Y,16)                                                      MD2-0350
      GO TO 105                                                          MD2-0360
145   CONTINUE                                                           MD2-0370
      CALL DATA (6)                                                      MD2-0380
      RETURN                                                             MD2-0390
C                                                                        MD2-0400
150   FORMAT (18H MOD2--GLOBAL VARS)                                     MD2-0410
155   FORMAT (A1)                                                        MD2-0420
160   FORMAT (30H TYPE IN MATRIX OR VECTOR NAME)                         MD2-0430
      END                                                                MD2-0440
```

BEST COPY AVAILABLE


```
      SUBROUTINE MOD3                                                    MD3-0010
      LOGICAL OLDFIL,PRT                                                 MD3-0020
      LOGICAL PUNCH                                                      MD3-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIMD3-0040
     1N,IPUNCH,PUNCH                                                     MD3-0050
      DIMENSION NAMES(2)                                                 MD3-0060
      COMMON U(38,5),X(49,6)                                             MD3-0070
      DATA NAMES/1HU,1HX/                                                MD3-0080
      DATA NX,NU,NZ,NPP/49,38,25,5/                                      MD3-0090
      DATA ICHARO/1HO/                                                   MD3-0100
      DATA ISIZU,ISIZX/38,49/                                            MD3-0110
      IF (PRT) WRITE (ICONO,140)                                         MD3-0120
      IF (OLDFIL) GO TO 105                                              MD3-0130
      GO TO 110                                                          MD3-0140
105   CALL DATA (7)                                                      MD3-0150
110   CONTINUE                                                           MD3-0160
115   IF (PRT) WRITE (ICONO,150)                                         MD3-0170
      READ (ICONI,145) INVAL                                             MD3-0180
      IF (PUNCH) WRITE (IPUNCH,145) INVAL                                MD3-0190
      IF (INVAL.EQ.ICHARO) GO TO 135                                     MD3-0200
      DO 120 I=1,2                                                       MD3-0210
        IF (INVAL.EQ.NAMES(I)) GO TO (125,130), I                        MD3-0220
120     CONTINUE                                                         MD3-0230
      GO TO 115                                                          MD3-0240
125   CALL QT (U,ISIZU,NU,NPP)                                           MD3-0250
      GO TO 115                                                          MD3-0260
130   CALL QT (X,ISIZX,NX,NPP+1)                                         MD3-0270
      GO TO 115                                                          MD3-0280
135   CALL DATA (8)                                                      MD3-0290
      RETURN                                                             MD3-0300
C                                                                        MD3-0310
140   FORMAT (25H MOD3--STATE,CONTROL VARS)                              MD3-0320
145   FORMAT (A1)                                                        MD3-0330
150   FORMAT (20H TYPE IN MATRIX NAME)                                   MD3-0340
      END                                                                MD3-0350
```

BEST COPY AVAILABLE


```
      SUBROUTINE MOD4                                                    MD4-0010
      LOGICAL PUNCH                                                      MD4-0020
      LOGICAL OLDFIL,PRT                                                 MD4-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICOND,ICONI,II MD4-0040
     1N,IPUNCH,PUNCH                                                     MD4-0050
      COMMON TRAN(12,16)                                                 MD4-0060
      DATA IMAX,JMAX/12,16/                                              MD4-0070
      DATA ISIZ/12/                                                      MD4-0080
      IF (PRT) WRITE (ICOND,120)                                         MD4-0090
      IF (OLDFIL) GO TO 110                                              MD4-0100
      DO 105 I=1,IMAX                                                    MD4-0110
        DO 105 J=1,JMAX                                                  MD4-0120
  105   TRAN(I,J)=0.0                                                    MD4-0130
      GO TO 115                                                          MD4-0140
  110 CALL DATA (3)                                                      MD4-0150
  115 CALL OT (TRAN,ISIZ,IMAX,JMAX)                                      MD4-0160
      CALL DATA (4)                                                      MD4-0170
      RETURN                                                             MD4-0180
C                                                                        MD4-0190
  120 FORMAT (21H MOD4-TRANSFER MATRIX)                                  MD4-0200
      END                                                                MD4-0210
```

BEST COPY AVAILABLE


```
      SUBROUTINE MD5                                                    MD5-0010
      LOGICAL OLDFIL,PRT                                                MD5-0020
      LOGICAL PUNCH                                                     MD5-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIMD5-0040
     1N,IPUNCH,PUNCH                                                    MD5-0050
      COMMON Q(45,5),VMS(80,5),Z(25,5)                                  MD5-0060
      DIMENSION NAMES(3)                                                MD5-0070
      DATA ISIZQ,ISIZV,ISIZZ/45,80,25/                                  MD5-0080
      DATA NUM/3/                                                       MD5-0090
      DATA NAMES/1HQ,1HV,1HZ/                                           MD5-0100
      DATA ICHARQ/1HQ/                                                  MD5-0110
      DATA NQ,NV,NZ,NPP/43,67,22,5/                                     MD5-0120
      IF (PRT) WRITE (ICONO,140)                                        MD5-0130
      IF (OLDFIL) CALL DATA (9)                                         MD5-0140
105   IF (PRT) WRITE (ICONO,145)                                        MD5-0150
      READ (ICONI,135) INVAL                                            MD5-0160
      IF (PUNCH) WRITE (IPUNCH,135) INVAL                               MD5-0170
      IF (INVAL.EQ.ICHARQ) GO TO 130                                    MD5-0180
      DO 110 I=1,NUM                                                    MD5-0190
        IF (INVAL.EQ.NAMES(I)) GO TO (115,120,125), I                   MD5-0200
110     CONTINUE                                                        MD5-0210
      GO TO 105                                                         MD5-0220
115   CALL QT (Q,ISIZQ,NQ,NPP)                                          MD5-0230
      GO TO 105                                                         MD5-0240
120   CALL QT (VMS,ISIZV,NV,NPP)                                        MD5-0250
      GO TO 105                                                         MD5-0260
125   CALL QT (Z,ISIZZ,NZ,NPP)                                          MD5-0270
      GO TO 105                                                         MD5-0280
130   CALL DATA (10)                                                    MD5-0290
      RETURN                                                            MD5-0300
C                                                                       MD5-0310
135   FORMAT (A1)                                                       MD5-0320
140   FORMAT (25H MOD5--Q AND VMS MATRICES)                             MD5-0330
145   FORMAT (20H TYPE IN MATRIX NAME)                                  MD5-0340
      END                                                               MD5-0350
```

BEST COPY AVAILABLE


```
      SUBROUTINE MOD6                                                    MD6-0010
      LOGICAL PUNCH                                                      MD6-0020
      LOGICAL OLDFIL,PRT                                                 MD6-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICOND,ICONI,IIMD6-0040
     1N,IPUNCH,PUNCH                                                     MD6-0050
      COMMON Q(45),VMS(80),Z(25)                                         MD6-0060
      DIMENSION NAMES(3)                                                 MD6-0070
      DATA NUM/3/                                                        MD6-0080
      DATA NAMES/1HQ,1HV,1HZ/                                            MD6-0090
      DATA NQ,NV,NZ/43,67,22/                                            MD6-0100
      DATA ICHARO/1HO/                                                   MD6-0110
      IF (PRT) WRITE (ICOND,135)                                         MD6-0120
      IWHICH=6                                                           MD6-0130
      IF (MOD(KEY-6,5).EQ.0) IWHICH=7                                    MD6-0140
      IF (OLDFIL) CALL DATA (11)                                         MD6-0150
  105 IF (PRT) WRITE (ICOND,140)                                         MD6-0160
      READ (ICONI,145) INVAL                                             MD6-0170
      IF (PUNCH) WRITE (IPUNCH,145) INVAL                                MD6-0180
      IF (INVAL.EQ.ICHARO) GO TO 130                                     MD6-0190
      DO 110 I=1,NUM                                                     MD6-0200
        IF (INVAL.EQ.NAMES(I)) GO TO (115,120,125), I                    MD6-0210
  110   CONTINUE                                                         MD6-0220
      GO TO 105                                                          MD6-0230
  115 CALL T (Q,NQ,IWHICH)                                               MD6-0240
      GO TO 105                                                          MD6-0250
  120 CALL T (VMS,NV,IWHICH)                                             MD6-0260
      GO TO 105                                                          MD6-0270
  125 CALL T (Z,NZ,IWHICH)                                               MD6-0280
      GO TO 105                                                          MD6-0290
  130 CALL DATA (12)                                                     MD6-0300
      RETURN                                                             MD6-0310
C                                                                        MD6-0320
  135 FORMAT (21H MOD6--UPDATE RECORDS)                                  MD6-0330
  140 FORMAT (20H TYPE IN MATRIX NAME)                                   MD6-0340
  145 FORMAT (A1)                                                        MD6-0350
      END                                                                MD6-0360
```

BEST COPY AVAILABLE


```
      SUBROUTINE DATA (I)                                                    DAT-0010
      LOGICAL PUNCH                                                          DAT-0020
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIDAT-0030
     1N,IPUNCH,PUNCH                                                         DAT-0040
      COMMON ONE                                                             DAT-0050
      DIMENSION INDEX(200)                                                   DAT-0060
      DIMENSION LENGTH(70)                                                   DAT-0070
      DATA LENGTH/67,5*1862,60*0,4*192/                                      DAT-0080
      GO TO (105,115,120,125,120,125,120,125,120,125,120,125,120,125), IDAT-0090
  105 CONTINUE                                                               DAT-0100
      CALL OPENMS (1,INDEX,200,0)                                            DAT-0110
      DO 110 I=7,62,5                                                        DAT-0120
        LENGTH(I)=1061                                                       DAT-0130
        LENGTH(I+1)=750                                                      DAT-0140
        LENGTH(I+2)=484                                                      DAT-0150
        LENGTH(I+3)=150                                                      DAT-0160
  110   LENGTH(I+4)=150                                                      DAT-0170
      RETURN                                                                 DAT-0180
  115 CONTINUE                                                               DAT-0190
      CALL CLOSMS (1,INDEX,200)                                              DAT-0200
      RETURN                                                                 DAT-0210
  120 CONTINUE                                                               DAT-0220
      CALL READMS (1,ONE,LENGTH(KEY),KEY)                                    DAT-0230
      RETURN                                                                 DAT-0240
  125 CONTINUE                                                               DAT-0250
      CALL WRITMS(1,ONE,LENGTH(KEY),KEY,0)                                   DAT-0260
      RETURN                                                                 DAT-0270
      END                                                                    DAT-0280
```

BEST COPY AVAILABLE


```
      SUBROUTINE S (ATRIX,IMAX)                                          SSS-0010
      LOGICAL PUNCH                                                      SSS-0020
      LOGICAL PRT,CLDFIL                                                 SSS-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICOND,ICONI,IISSS-0040
     1N,IPUNCH,PUNCH                                                     SSS-0050
      DIMENSION ATRIX(1)                                                 SSS-0060
100   IF (PRT) WRITE (ICONO,115)                                         SSS-0070
C     NON-ANSI UNFORMATTED READ                                          SSS-0080
      READ , I,VAL                                                       SSS-0090
      IF (PUNCH) WRITE (IPUNCH,125) I,VAL                                SSS-0100
      IF (I.EQ.0) RETURN                                                 SSS-0110
      IF (I.LE.IMAX) GO TO 110                                           SSS-0120
      WRITE (ICONO,120)                                                  SSS-0130
      GO TO 105                                                          SSS-0140
110   ATRIX(I)=VAL                                                       SSS-0150
      GO TO 105                                                          SSS-0160
C                                                                        SSS-0170
115   FORMAT (9H  I   VAL)                                               SSS-0180
120   FORMAT (21H BAD INDEX--TRY AGAIN)                                  SSS-0190
125   FORMAT (I3,3X,E13.4)                                               SSS-0200
      END                                                                SSS-0210
```

BEST COPY AVAILABLE


```
      SUBROUTINE QT (ATRIX,ISIZD,IMAX,JMAX)                                 QT -0010
      DIMENSION ATRIX(ISIZD,JMAX)                                           QT -0020
      LOGICAL PUNCH                                                         QT -0030
      LOGICAL PRT,OLDFIL                                                    QT -0040
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IIQT -0050
     1N,IPUNCH,PUNCH                                                        QT -0060
105   IF (PRT) WRITE (ICONO,115)                                            QT -0070
C     NON-ANSI UNFORMATTED READ                                             QT -0080
      READ , I,J,VAL                                                        QT -0090
      IF (PUNCH) WRITE (IPUNCH,125) I,J,VAL                                 QT -0100
      IF (I.EQ.0) RETURN                                                    QT -0110
      IF (I.LE.IMAX.AND.J.LE.JMAX) GO TO 110                                QT -0120
      WRITE (ICONO,120)                                                     QT -0130
      GO TO 105                                                             QT -0140
110   ATRIX(I,J)=VAL                                                        QT -0150
      GO TO 105                                                             QT -0160
C                                                                           QT -0170
115   FORMAT (11H   I   J   VAL)                                            QT -0180
120   FORMAT (23H BAD INDICES--TRY AGAIN)                                   QT -0190
125   FORMAT (2I3,E13.4)                                                    QT -0200
      END                                                                   QT -0210
```

BEST COPY AVAILABLE


```
      SUBROUTINE T(ATRIX,IMAX,JMAX)                                          TTT-0010
      LOGICAL PUNCH                                                          TTT-0020
      LOGICAL PRT,OLDFIL                                                     TTT-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,IS9ZE,KEY,ICONO,ICONI,IITTT-0040
     1N,IPUNCH,PUNCH                                                         TTT-0050
      DIMENSION ATRIX(IMAX)                                                  TTT-0060
105   IF (PRT) WRITE (ICONO,115)                                             TTT-0070
C     NON-ANSI UNFORMATTED READ                                              TTT-0080
      READ , I,J,VAL                                                         TTT-0090
      IF (PUNCH) WRITE (IPUNCH,125) I,J,VAL                                  TTT-0100
      IF (I.EQ.0) RETURN                                                     TTT-0110
      IF (I.LE.IMAX.AND.J.EQ.JMAX) GO TO 110                                 TTT-0120
      WRITE (ICONO,120)                                                      TTT-0130
      GO TO 105                                                              TTT-0140
110   ATRIX(I)=VAL                                                           TTT-0150
      GO TO 105                                                              TTT-0160
C                                                                            TTT-0170
115   FORMAT (11H  I  J  VAL)                                                TTT-0180
120   FORMAT (23H BAD INDICES--TRY AGAIN)                                    TTT-0190
125   FORMAT (2I3,E13.4)                                                     TTT-0200
      END                                                                    TTT-0210
```

BEST COPY AVAILABLE


```
      SUBROUTINE SPEC (IROW,ICOL,ATRIX,IMAX,JMAX,MPTR,TRIX)                SPC-0010
      LOGICAL PUNCH                                                        SPC-0020
      LOGICAL OLDFIL,PRT                                                   SPC-0030
      COMMON /CONTRL/ OLDFIL,PRT,IRTPS(70),IPTR,ISIZE,KEY,ICONO,ICONI,IISPC-0040
     1N,IPUNCH,PUNCH                                                       SPC-0050
      DIMENSION TRIX(IMAX,JMAX), IROW(MPTR), ICOL(MPTR), ATRIX(MPTR)       SPC-0060
      DO 105 I=1,IMAX                                                      SPC-0070
        DO 105 J=1,JMAX                                                    SPC-0080
105     TRIX(I,J)=0.0                                                      SPC-0090
      IF (.NOT.OLDFIL) GO TO 115                                           SPC-0100
      IF (MPTR.EQ.0) GO TO 115                                             SPC-0110
      DO 110 I=1,MPTR                                                      SPC-0120
        IVAL=IROW(I)                                                       SPC-0130
        JVAL=ICOL(I)                                                       SPC-0140
110     TRIX(IVAL,JVAL)=ATRIX(I)                                           SPC-0150
115   CONTINUE                                                             SPC-0160
120   IF (PRT) WRITE (ICONO,140)                                           SPC-0170
C     NON-ANSI UNFORMATTED READ                                            SPC-0180
      READ , I,J,VAL                                                       SPC-0190
      IF (PUNCH) WRITE (IPUNCH,150) I,J,VAL                                SPC-0200
      IF (I.EQ.0) GO TO 125                                                SPC-0210
      IF (I.GT.IMAX.OR.J.GT.JMAX) GO TO 135                                SPC-0220
      TRIX(I,J)=VAL                                                        SPC-0230
      GO TO 120                                                            SPC-0240
125   MPTR=0                                                               SPC-0250
      DO 130 I=1,IMAX                                                      SPC-0260
        DO 130 J=1,JMAX                                                    SPC-0270
        IF (TRIX(I,J).EQ.0) GO TO 130                                      SPC-0280
        MPTR=MPTR+1                                                        SPC-0290
        ATRIX(MPTR)=TRIX(I,J)                                              SPC-0300
        IROW(MPTR)=I                                                       SPC-0310
        ICOL(MPTR)=J                                                       SPC-0320
130     CONTINUE                                                           SPC-0330
      RETURN                                                               SPC-0340
135   WRITE (ICONO,145)                                                    SPC-0350
      GO TO 120                                                            SPC-0360
C                                                                          SPC-0370
140   FORMAT (11H  I  J  VAL)                                              SPC-0380
145   FORMAT (23H BAD INDICES--TRY AGAIN)                                  SPC-0390
150   FORMAT (2I3,E13.4)                                                   SPC-0400
      END                                                                  SPC-0410
```

BEST COPY AVAILABLE

CHAPTER V


```
      PROGRAM VIEW(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5=INPUT,TAPE1)          VUU-0010
C                                                                        VUU-0020
C                                                                        VUU-0030
C        ////VERSION 1.0--FEBRUARY 1973                                  VUU-0040
C                                                                        VUU-0050
C     PART OF THE NATIONAL PLANNING MODEL PROJECT FOR NCHEMS AT          VUU-0060
C     WICHE, PREPARED UNDER USOE CONTRACT.                               VUU-0070
C                                                                        VUU-0080
C                     OEC-0-72-3575                                      VUU-0090
C                                                                        VUU-0100
C                                                                        VUU-0110
C        ***NOTE***      NO WARRANTY, EITHER EXPRESS OR IMPLIED,         VUU-0120
C                        IS MADE FOR THIS COLLECTION OF PROGRAMS         VUU-0130
C                        OR FOR ITS ACCOMPANYING TEST DATA.              VUU-0140
C                        ALL CURRENT SOFTWARE FOR THIS PROJECT IS        VUU-0150
C                        RELEASED AS NCHEMS TYPE II SOFTWARE,            VUU-0160
C                        WITH NO PROVISION MADE FOR PROGRAM ACCURACY     VUU-0170
C                        OR MAINTENANCE.                                 VUU-0180
C                                                                        VUU-0190
C     THIS PROGRAM IS A GENERALIZED SUMMARY DEVICE INTENDED FOR ON-LINE  VUU-0200
C     INTERACTIVE USE WITH THE NCHEMS NATIONAL PLANNING MODEL PROJECT.   VUU-0210
C     BASICLY, THE PROGRAM SELECTS (UNDER USER CONTROL) VARIOUS INFORM-  VUU-0220
C     ATION FROM A RANDOM ACCESS FILE (TAPE1)  WHICH HAS BEEN            VUU-0230
C     INITIALIZED BY THE MODEL ITSELF (NPM).  ONLY VERY SELECTED INFO    VUU-0240
C     CAN BE REQUESTED--FOR MORE EXTENSIVE DATA THE USER SHOULD CONSULT  VUU-0250
C     NPM PRINTER DUMP.                                                  VUU-0260
C                                                                        VUU-0270
C     VIEW* BEARS THREE NON-ANSI QUALITIES.                              VUU-0280
C        1)  THE TELETYPE  INPUT (UNFORMATTED)                           VUU-0290
C        2)  THE ACCESSING METHOD FOR RANDOM-ACCESS FILES                VUU-0300
C        3)  THE SPECIAL CODES USED TO INVOKE STANDARD ASCII             VUU-0310
C                 CONTROL CHARACTERS FOR CRT DISPLAY.                    VUU-0320
C                                                                        VUU-0330
C     VERSION 1.0 IS CDC 6000--KRONOS DEPENDENT IN ALL THREE CATEGORIES. VUU-0340
C                                                                        VUU-0350
C                                                                        VUU-0360
      COMMON /FILES/ ICONI,ICONO,IDUMP                                   VUU-0370
      COMMON ENROL1(6,17),DIF1(6,17),FTSTU1(6,17),STATS1(6,6)            VUU-0380
      COMMON ENROL2(6,17),DIF2(6,17),FTSTU2(6,17),STATS2(6,6)            VUU-0390
      COMMON PIC1(6,6),PIC2(6,6),TOTAL1(6),TOTAL2(6)                     VUU-0400
      COMMON NDICT(20,10),INDEX(40),IND1,IND2                            VUU-0410
      DIMENSION E1(342), E2(342)                                         VUU-0420
      EQUIVALENCE (E1,ENROL1), (E2,ENROL2)                               VUU-0430
C                                                                        VUU-0440
C     CRT SENSITIVE CHARACTERS---SET TO BLANKS HERE                      VUU-0450
      DATA FE,HWOCL/2H  ,2H  /                                           VUU-0460
C                                                                        VUU-0470
C                                                                        VUU-0480
C     INITIALIZE RANDOM ACCESS FILE, UNIT 1.                             VUU-0490
      ICONI=5                                                            VUU-0500
      ICONO=6                                                            VUU-0510
      IDUMP=1                                                            VUU-0520
```

BEST COPY AVAILABLE


```
      CALL OPENMS (IDUMP,INDEX,40,0)                                    VUU-0530
C                                                                       VUU-0540
C     READ IN THE CURRENT DIRECTORY FOR FINANCING ALTERNATIVES --       VUU-0550
C     LENGTH 181                                                        VUU-0560
      CALL READMS (IDUMP,NDICT,200,1)                                   VUU-0570
C                                                                       VUU-0580
C     PUT OUT FIRST FRAME                                               VUU-0590
110   WRITE (ICONO,340) FF                                              VUU-0600
C                                                                       VUU-0610
C     NON-ANSI-UNFORMATTED READ                                         VUU-0620
      READ , IND1                                                       VUU-0630
C                                                                       VUU-0640
      IF (IND1.NE.0) GO TO 130                                          VUU-0650
      DO 120 I=1,342                                                    VUU-0660
120     E1(I)=RANF(0)*1000.0                                            VUU-0670
      GO TO 150                                                         VUU-0680
130   CONTINUE                                                          VUU-0690
      IF (NDICT(IND1,1).NE.0) GO TO 140                                 VUU-0700
      WRITE (ICONO,350) FF                                              VUU-0710
      GO TO 110                                                         VUU-0720
140   INX1=IND1+1                                                       VUU-0730
      WRITE (ICONO,360) IND1,(NDICT(IND1,J),J=1,10)                     VUU-0740
      CALL READMS (IDUMP,ENROL1,342,INX1)                               VUU-0750
150   CONTINUE                                                          VUU-0760
      WRITE (ICONO,370)                                                 VUU-0770
C     NON-ANSI-UNFORMATTED READ                                         VUU-0780
      READ , IND2                                                       VUU-0790
      IF (IND2.NE.0) GO TO 170                                          VUU-0800
      DO 160 I=1,342                                                    VUU-0810
160     E2(I)=RANF(0)*1000.0                                            VUU-0820
      GO TO 190                                                         VUU-0830
170   CONTINUE                                                          VUU-0840
      IF (NDICT(IND2,1).NE.0) GO TO 180                                 VUU-0850
      WRITE (ICONO,350) FF                                              VUU-0860
      GO TO 150                                                         VUU-0870
180   INX2=IND2+1                                                       VUU-0880
      WRITE (ICONO,360) IND2,(NDICT(IND2,J),J=1,10)                     VUU-0890
      CALL READMS (IDUMP,ENROL2,342,INX2)                               VUU-0900
190   CONTINUE                                                          VUU-0910
C                                                                       VUU-0920
C     DATA NOW LOADED                                                   VUU-0930
C     PROCEED WITH THE MENU                                             VUU-0940
200   CONTINUE                                                          VUU-0950
      WRITE(ICONO,510)                                                  VUU-0960
      WRITE (ICONO,380) FF                                              VUU-0970
      WRITE(ICONO,510)                                                  VUU-0980
C     NON-ANSI-UNFORMATTED READ                                         VUU-0990
210   READ , ISWIT                                                      VUU-1000
      IF (ISWIT.LT.0.OR.ISWIT.GT.11) GO TO 220                          VUU-1010
      IF (ISWIT.NE.0) GO TO (230,260,270,280,290,300,310,320,330,200,110VUU-1020
     1), ISWIT                                                          VUU-1030
      WRITE (ICONO,390) FF                                              VUU-1040
```

BEST COPY AVAILABLE


```
      STOP                                                              VUU-1050
220   WRITE (ICONO,400) FF                                              VUU-1060
      GO TO 200                                                         VUU-1070
C                                                                       VUU-1080
C     PROCESS SUMMARY OF INSTITUTIONAL STATISTICS                       VUU-1090
230   CONTINUE                                                          VUU-1100
      DO 240 I=1,6                                                      VUU-1110
        TOTAL1(I)=0.0                                                   VUU-1120
240     TOTAL2(I)=0.0                                                   VUU-1130
      DO 250 J=1,5                                                      VUU-1140
        DO 250 I=1,6                                                    VUU-1150
        TOTAL1(J)=TOTAL1(J)+STATS1(I,J)                                 VUU-1160
250     TOTAL2(J)=TOTAL2(J)+STATS2(I,J)                                 VUU-1170
      TOTAL1(6)=0.0                                                     VUU-1180
      TOTAL2(6)=0.0                                                     VUU-1190
      WRITE (ICONO,410) FF,IND1                                         VUU-1200
      WRITE (ICONO,420) ((STATS1(I,J),J=1,6),I=1,6),TOTAL1,IND2,((STATS2VUU-1210
     1(I,J),J=1,6),I=1,6),TOTAL2                                        VUU-1220
      GO TO 210                                                         VUU-1230
C                                                                       VUU-1240
C     PROCESS TOTAL STUDENTS ENROLLED BY ABILITY QUARTILE               VUU-1250
C                                                                       VUU-1260
260   CONTINUE                                                          VUU-1270
      WRITE (ICONO,430) FF,IND1                                         VUU-1280
      CALL SUBA (ENROL1,ENROL2,1.0,0)                                   VUU-1290
      GO TO 210                                                         VUU-1300
C                                                                       VUU-1310
C     PROCESS TOTAL STUDENTS ENROLLED BY ABILITY QUARTILE               VUU-1320
C                                                                       VUU-1330
270   CONTINUE                                                          VUU-1340
      WRITE (ICONO,440) FF,IND1                                         VUU-1350
      CALL SUBI (ENROL1,ENROL2,1.0,0)                                   VUU-1360
      GO TO 210                                                         VUU-1370
C                                                                       VUU-1380
C     PROCESS UNUSED STUDENT SPACES BY ABILITY QUARTILE                 VUU-1390
C                                                                       VUU-1400
280   CONTINUE                                                          VUU-1410
      WRITE (ICONO,450) FF,IND1                                         VUU-1420
      CALL SUBA (DIF1,DIF2,1.0,1)                                       VUU-1430
      GO TO 210                                                         VUU-1440
C                                                                       VUU-1450
C     PROCESS UNUSED STUDENT SPACES BY INCOME QUARTILE                  VUU-1460
290   CONTINUE                                                          VUU-1470
C                                                                       VUU-1480
      WRITE (ICONO,460) FF,IND1                                         VUU-1490
      CALL SUBI (DIF1,DIF2,1.0,1)                                       VUU-1500
      GO TO 210                                                         VUU-1510
C                                                                       VUU-1520
C     PROCESS UNSATISFIED STUDENT DEMAND BY ABILITY QUARTILE            VUU-1530
C                                                                       VUU-1540
300   CONTINUE                                                          VUU-1550
      WRITE (ICONO,470) FF,IND1                                         VUU-1560
```

BEST COPY AVAILABLE


```
      CALL SUBA (DIF1,DIF2,-1.0,1)                                      VUU-1570
      GO TO 210                                                         VUU-1580
C                                                                       VUU-1590
C     PROCESS UNSATISFIED STUDENT DEMAND BY INCOME LEVEL                VUU-1600
C                                                                       VUU-1610
310   CONTINUE                                                          VUU-1620
      WRITE (ICONO,480) FF,IND1                                         VUU-1630
      CALL SUB1 (DIF1,DIF2,-1.0,1)                                      VUU-1640
      GO TO 210                                                         VUU-1650
C                                                                       VUU-1660
C     PROCESS FIRST TIME STUDENTS BY ABILITY QUARTILE                   VUU-1670
C                                                                       VUU-1680
320   CONTINUE                                                          VUU-1690
      WRITE (ICONO,490) FF,IND1                                         VUU-1700
      CALL SUBA (FTSTU1,FTSTU2,1.0,0)                                   VUU-1710
      GO TO 210                                                         VUU-1720
C     PROCESS FIRST TIME STUDENTS BY INCOME QUARTILE                    VUU-1730
C                                                                       VUU-1740
330   CONTINUE                                                          VUU-1750
      WRITE (ICONO,500) FF,IND1                                         VUU-1760
      CALL SUB1 (FTSTU1,FTSTU2,1.0,0)                                   VUU-1770
      GO TO 210                                                         VUU-1780
C                                                                       VUU-1790
340   FORMAT (A2,20X,30HVIEW--VERSION 2.0--MARCH, 1973,//,60H ENTER INDIVUU-1800
     1CES OF ALTERNATIVE FINANCING PLANS TO BE COMPARED,/,10X,53H ---A ZVUU-1810
     2ERO INDEX FOR EITHER ENTRY REQUESTS DUMMY DATA,//,1X,21H INDEX FORVUU-1820
     3 FIRST PLAN)                                                      VUU-1830
350   FORMAT (R2,33H BAD OR NON-EXISTENT INDEX--RETRY)                  VUU-1840
360   FORMAT (23H HEADER FOR ALTERNATIVE,I3,6H IS...,/1X,10A6)          VUU-1850
370   FORMAT (22H INDEX FOR SECOND PLAN)                                VUU-1860
380   FORMAT (A2,31H MENU OF ACTIONS AND REPORTS...,/,10X,29H0 - ALWAYS VUU-1870
     1TERMINATES PROGRAM,/,10X,50H1 - REPORT 1 - SUMMARY OF INSTITUTIONAVUU-1880
     2L STATISTICS,/,10X,58H2 - REPORT 2 - TOTAL STUDENTS ENROLLED BY ABVUU-1890
     3ILITY QUARTILE,/,10X,57H3 - REPORT 3 - TOTAL STUDENTS ENROLLED BY VUU-1900
     4INCOME QUARTILE,/,10X,56H4 - REPORT 4 - UNUSED STUDENT SPACES BY AVUU-1910
     5BILITY QUARTILE,/,10X,55H5 - REPORT 5 - UNUSED STUDENT SPACES BY IVUU-1920
     6NCOME QUARTILE,/,10X,61H6 - REPORT 6 - UNSATISFIED STUDENT DEMAND VUU-1930
     7BY ABILITY QUARTILE,/,10X,60H7 - REPORT 7 - UNSATISFIED STUDENT DEVUU-1940
     8MAND BY INCOME QUARTILE,/,10X,54H8 - REPORT 8 - FIRST TIME STUDENTVUU-1950
     9S BY ABILITY QUARTILE,/,10X,53H9 - REPORT 9 - FIRST TIME STUDENTS VUU-1960
     $BY INCOME QUARTILE,/,10X,39H10- ALWAYS RETURNS THIS LIST OF REPORTVUU-1970
     $S,/,10X,41H11- ALLOWS USER TO ENTER NEW PLAN INDICES,/,40H ENTER IVUU-1980
     $NDEX OF ACTION OR REPORT DESIRED)                                 VUU-1990
390   FORMAT (A2,22H TERMINATION REQUESTED)                             VUU-2000
400   FORMAT (A2,21H BAD INDEX--TRY AGAIN)                              VUU-2010
410   FORMAT (A2,7X,8H****PLAN,I3,47H  --SUMMARY OF 1974 INSTITUTIONAL SVUU-2020
     1TATISTICS---)                                                     VUU-2030
420   FORMAT (15H   PUBLIC  UNIV,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H   PUBVUU-2040
     1LIC  4-YR,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H   PUBLIC  2-YR,F11.0,FVUU-2050
     28.0,2F9.0,F11.0,F8.2,/,15H   PRIVATE UNIV,F11.0,F8.0,2F9.0,F11.0,FVUU-2060
     38.2,/,15H   PRIVATE 4-YR,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H   PRIVAVUU-2070
     4TE 2-YR,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H    --TOTAL--  ,F11.0,F8.VUU-2080
```

BEST COPY AVAILABLE


```
     50,2F9.0,F11.0,F8.2,/,7X,8H****PLAN,I3,54H   NCB    FACULTY   SPACEVUU-2090
     6   STUDENTS    FEE $    $/STU,/,15H   PUBLIC  UNIV,F11.0,F8.0,2F9.VUU-2100
     70,F11.0,F8.2,/,15H   PUBLIC  4-YR,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15VUU-2110
     8H   PUBLIC  2-YR,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H   PRIVATE UNIV,VUU-2120
     9F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H   PRIVATE 4-YR,F11.0,F8.0,2F9.0,VUU-2130
     $F11.0,F8.2,/,15H   PRIVATE 2-YR,F11.0,F8.0,2F9.0,F11.0,F8.2,/,15H VUU-2140
     $   --TOTAL--  ,F11.0,F8.0,2F9.0,F11.0,F8.2,//,45H ENTER INDEX OF NVUU-2150
     $EXT ACTION OR REPORT DESIRED,///)                                  VUU-2160
430   FORMAT (A2,7X,8H****PLAN,I3,49H  --TOTAL STUDENTS ENROLLED BY ABILVUU-2170
     1ITY QUARTILE--)                                                    VUU-2180
440   FORMAT (A2,7X,8H****PLAN,I3,48H  --TOTAL STUDENTS ENROLLED BY INCOVUU-2190
     1ME QUARTILE--)                                                     VUU-2200
450   FORMAT (A2,7X,8H****PLAN,I3,47H  --UNUSED STUDENT SPACES BY ABILITVUU-2210
     1Y QUARTILE--)                                                      VUU-2220
460   FORMAT (A2,7X,8H****PLAN,I3,46H  --UNUSED STUDENT SPACES BY INCOMEVUU-2230
     1 QUARTILE--)                                                       VUU-2240
470   FORMAT (A2,7X,8H****PLAN,I3,50H  --UNSATISFIED STUDENT DEMAND BY AVUU-2250
     1BILITY QUARTILE)                                                   VUU-2260
480   FORMAT (A2,7X,8H****PLAN,I3,49H  UNSATISFIED STUDENT DEMAND BY INCVUU-2270
     1OME QUARTILE--)                                                    VUU-2280
490   FORMAT (A2,7X,8H****PLAN,I3,45H  --FIRST-TIME STUDENTS BY ABILITY VUU-2290
     1QUARTILE--)                                                        VUU-2300
500   FORMAT (A2,7X,8H****PLAN,I3,44H  --FIRST-TIME STUDENTS BY INCOME QVUU-2310
     1UARTILE--)                                                         VUU-2320
510   FORMAT(///)                                                       VUU-2330
      END                                                               VUU-2340
```

BEST COPY AVAILABLE


```
      SUBROUTINE SUBA (AR1,AR2,SIGN,IFLAG)                                  SBA-0010
      DIMENSION AR1(6,17), AR2(6,17)                                        SBA-0020
C                                                                           SBA-0030
      COMMON /FILES/ ICONI,ICONO,IDUMP                                      SBA-0040
      COMMON ENROL1(6,17),DIF1(6,17),FTSTU1(6,17),STATS1(6,6)               SBA-0050
      COMMON ENROL2(6,17),DIF2(6,17),FTSTU2(6,17),STATS2(6,6)               SBA-0060
      COMMON PIC1(6,6),PIC2(6,6),TOTAL1(6),TOTAL2(6)                        SBA-0070
      COMMON NDICT(20,10),INDEX(40),IND1,IND2                               SBA-0080
C                                                                           SBA-0090
C                                                                           SBA-0100
      DO 120 I=1,6                                                          SBA-0110
        K=0                                                                 SBA-0120
        DO 120 J=1,4                                                        SBA-0130
        TEMP1=0.0                                                           SBA-0140
        TEMP2=0.0                                                           SBA-0150
        DO 110 M=1,4                                                        SBA-0160
          K=K+1                                                             SBA-0170
          TEMP1=TEMP1+AR1(I,K)                                              SBA-0180
110       TEMP2=TEMP2+AR2(I,K)                                              SBA-0190
        PIC1(I,J)=TEMP1*SIGN                                                SBA-0200
120     PIC2(I,J)=TEMP2*SIGN                                                SBA-0210
      DO 130 I=1,6                                                          SBA-0220
        PIC1(I,5)=AR1(I,17)*SIGN                                            SBA-0230
130     PIC2(I,5)=AR2(I,17)*SIGN                                            SBA-0240
      DO 150 I=1,6                                                          SBA-0250
        TEMP1=0.0                                                           SBA-0260
        TEMP2=0.0                                                           SBA-0270
        DO 140 J=1,5                                                        SBA-0280
          TEMP1=TEMP1+PIC1(I,J)                                             SBA-0290
140       TEMP2=TEMP2+PIC2(I,J)                                             SBA-0300
        PIC1(I,6)=TEMP1                                                     SBA-0310
150     PIC2(I,6)=TEMP2                                                     SBA-0320
      IF (IFLAG.EQ.0) GO TO 170                                             SBA-0330
      DO 160 I=1,6                                                          SBA-0340
        DO 160 J=1,6                                                        SBA-0350
        IF (PIC1(I,J).LT.0.0) PIC1(I,J)=0.0                                 SBA-0360
160     IF (PIC2(I,J).LT.0.0) PIC2(I,J)=0.0                                 SBA-0370
170   CONTINUE                                                              SBA-0380
      DO 190 J=1,6                                                          SBA-0390
        TEMP1=0.0                                                           SBA-0400
        TEMP2=0.0                                                           SBA-0410
        DO 180 I=1,6                                                        SBA-0420
          TEMP1=TEMP1+PIC1(I,J)                                             SBA-0430
180       TEMP2=TEMP2+PIC2(I,J)                                             SBA-0440
        TOTAL1(J)=TEMP1                                                     SBA-0450
190     TOTAL2(J)=TEMP2                                                     SBA-0460
      WRITE (ICONO,200) ((PIC1(I,J),J=1,6),I=1,6),TOTAL1,IND2,((PIC2(I,JSBA-0470
     1),J=1,6),I=1,6),TOTAL2                                              SBA-0480
      RETURN                                                                SBA-0490
C                                                                           SBA-0500
C                                                                           SBA-0510
C                                                                           SBA-0520
```

BEST COPY AVAILABLE


```
200   FORMAT (17H   PUBLIC  UNIV  ,5F9.0,F10.0,/,17H   PUBLIC  4-YR  ,5FSBA-0530
     19.0,F10.0,/,17H   PUBLIC  2-YR  ,5F9.0,F10.0,/,17H   PRIVATE UNIV SBA-0540
     2 ,5F9.0,F10.0,/,17H   PRIVATE 4-YR  ,5F9.0,F10.0,/,17H   PRIVATE 2SBA-0550
     3-YR  ,5F9.0,F10.0,/,17H    --TOTAL--     ,5F9.0,F10.0,/,7X,8H****PLSBA-0560
     4AN,I3,53H  FIRST    SECOND    THIRD    FOURTH    GRAD    TOTAL,/,1SBA-0570
     57H   PUBLIC  UNIV  ,5F9.0,F10.0,/,17H   PUBLIC  4-YR  ,5F9.0,F10.0SBA-0580
     6,/,17H   PUBLIC  2-YR  ,5F9.0,F10.0,/,17H   PRIVATE UNIV  ,5F9.0,FSBA-0590
     710.0,/,17H   PRIVATE 4-YR  ,5F9.0,F10.0,/,17H   PRIVATE 2-YR  ,5F9SBA-0600
     8.0,F10.0,/,17H    --TOTAL--     ,5F9.0,F10.0,//,40H ENTER INDEX OF SBA-0610
     9ACTION OR REPORT DESIRED,///)                                      SBA-0620
      END                                                                SBA-0630
```

BEST COPY AVAILABLE


```
      SUBROUTINE SUBI (AR1,AR2,SIGN,IFLAG)                                  SBI-0010
      DIMENSION AR1(6,17)                                                   SBI-0020
      DIMENSION AR2(6,17)                                                   SBI-0030
C                                                                           SBI-0040
      COMMON /FILES/ ICONI,ICONO,IDUMP                                      SBI-0050
      COMMON ENROL1(6,17),DIF1(6,17),FTSTU1(6,17),STATS1(6,6)               SBI-0060
      COMMON ENROL2(6,17),DIF2(6,17),FTSTU2(6,17),STATS2(6,6)               SBI-0070
      COMMON PIC1(6,6),PIC2(6,6),TOTAL1(6),TOTAL2(6)                        SBI-0080
      COMMON NDICT(20,10),INDEX(40),IND1,IND2                               SBI-0090
C                                                                           SBI-0100
C                                                                           SBI-0110
      DO 120 I=1,6                                                          SBI-0120
        DO 120 J=1,4                                                        SBI-0130
        TEMP1=0.0                                                           SBI-0140
        TEMP2=0.0                                                           SBI-0150
        K=J                                                                 SBI-0160
        DO 110 M=1,4                                                        SBI-0170
          TEMP1=TEMP1+AR1(I,K)                                              SBI-0180
          TEMP2=TEMP2+AR2(I,K)                                              SBI-0190
110       K=K+4                                                             SBI-0200
        PIC1(I,J)=TEMP1*SIGN                                                SBI-0210
120     PIC2(I,J)=TEMP2*SIGN                                                SBI-0220
      DO 130 I=1,6                                                          SBI-0230
        PIC1(I,5)=AR1(I,17)*SIGN                                            SBI-0240
130     PIC2(I,5)=AR2(I,17)*SIGN                                            SBI-0250
      DO 150 I=1,6                                                          SBI-0260
        TEMP1=0.0                                                           SBI-0270
        TEMP2=0.0                                                           SBI-0280
        DO 140 J=1,5                                                        SBI-0290
          TEMP1=TEMP1+PIC1(I,J)                                             SBI-0300
140       TEMP2=TEMP2+PIC2(I,J)                                             SBI-0310
        PIC1(I,6)=TEMP1                                                     SBI-0320
150     PIC2(I,6)=TEMP2                                                     SBI-0330
      IF (IFLAG.EQ.0) GO TO 170                                             SBI-0340
      DO 160 I=1,6                                                          SBI-0350
        DO 160 J=1,6                                                        SBI-0360
        IF (PIC1(I,J).LT.0.0) PIC1(I,J)=0.0                                 SBI-0370
160     IF (PIC2(I,J).LT.0.0) PIC2(I,J)=0.0                                 SBI-0380
170   CONTINUE                                                              SBI-0390
      DO 190 J=1,6                                                          SBI-0400
        TEMP1=0.0                                                           SBI-0410
        TEMP2=0.0                                                           SBI-0420
        DO 180 I=1,6                                                        SBI-0430
          TEMP1=TEMP1+PIC1(I,J)                                             SBI-0440
180       TEMP2=TEMP2+PIC2(I,J)                                             SBI-0450
        TOTAL1(J)=TEMP1                                                     SBI-0460
190     TOTAL2(J)=TEMP2                                                     SBI-0470
      WRITE (ICONO,200) ((PIC1(I,J),J=1,6),I=1,6),TOTAL1,IND2,((PIC2(I,JSBI-0480
     1),J=1,6),I=1,6),TOTAL2                                              SBI-0490
      RETURN                                                                SBI-0500
C                                                                           SBI-0510
C                                                                           SBI-0520
```

BEST COPY AVAILABLE


```
C                                                                       SBI-0530
200   FORMAT (17H   PUBLIC UNIV  ,5F9.0,F10.0,/,17H   PUBLIC 4-YR  ,5FSBI-0540
     19.0,F10.0,/,17H   PUBLIC 2-YR  ,5F9.0,F10.0,/,17H   PRIVATE UNIV SBI-0550
     2 ,5F9.0,F10.0,/,17H   PRIVATE 4-YR  ,5F9.0,F10.0,/,17H   PRIVATE 2SBI-0560
     3-YR  ,5F9.0,F10.0,/,17H    --TOTAL--     ,5F9.0,F10.0,/,7X,8H****PLSBI-0570
     4AN,13,53H  FIRST    SECOND    THIRD    FOURTH    GRAD    TOTAL,/,1SBI-0580
     57H   PUBLIC UNIV  ,5F9.0,F10.0,/,17H   PUBLIC 4-YR  ,5F9.0,F10.0SBI-0590
     6,/,17H   PUBLIC 2-YR  ,5F9.0,F10.0,/,17H   PRIVATE UNIV  ,5F9.0,FSBI-0600
     710.0,/,17H   PRIVATE 4-YR  ,5F9.0,F10.0,/,17H   PRIVATE 2-YR  ,5F9SBI-0610
     8.0,F10.0,/,17H    --TOTAL--     ,5F9.0,F10.0,//,40H ENTER INDEX OF SBI-0620
     9ACTION OR REPORT DESIRED,///)                                     SBI-0630
      END                                                               SBI-0640
```

BEST COPY AVAILABLE

CHAPTER VI

DESCRIPTION OF OTHER RELATED REPORTS

Additional information related to the results of this research effort can be obtained from other National Planning Model - Phase II project reports. A listing of all of the project reports would include:

1. A Federal Planning Model for Analysis of Accessibility to Higher Education: An Overview. A summary document that presents a discussion of the prototype model in nontechnical terms such that the basic concepts can be understood by the higher education community. This includes a set of example calculations to illustrate the computations in the model.
2. A Design for a Federal Planning Model for Analysis of Accessibility to Higher Education. A documentation of the assumptions, design considerations, detailed prototype model relationships, and possible future research. This includes the most detailed explanation currently available for the prototype model.
3. Prototype Software for a Federal Planning Model for Analysis of Accessibility to Higher Education. A complete listing of the prototype software for:

BEST COPY AVAILABLE

a. MODIFY -- a routine that creates or updates the data base for the prototype model.

b. NPM 2.4 -- the current version of the prototype model.

c. VIEW -- output report routine that displays several summary reports from runs of the model comparing two alternative financing plans.

4. Preliminary Operating Instructions for a Federal Planning Model for Analysis of Accessibility to Higher Education. A report that presents very preliminary instructions for using the current prototype model software. This report is not a general user's manual as it does require extensive knowledge of the model and the software. It does, however, provide an initial set of instructions that can be used with the prototype and a basis for an improved user's manual in the future.

5. Preliminary Data for a Federal Planning Model for Analysis of Accessibility to Higher Education. A preliminary report to illustrate the types of data used in preliminary tests of the prototype model. This report contains all of the prototype data values, description of each variable, and the current source of the data.

6. Preliminary Test Reports from a Federal Planning Model for Analysis of Accessibility to Higher Education. A complete set of the current output reports illustrating the current

BEST COPY AVAILABLE

operational status of the prototype model. Included are the summary output reports comparing two alternative financing plans and a complete step-by-step report of the status of the model at a number of intermediate checkpoints in the model operation. The step-by-step report includes both a simulation run of the institutional sector of the model and a segment of an optimization run illustrating improvements in objective function values.

All of the above reports should be considered preliminary reports on the National Planning Models effort by NCHEMS. These reports should and will be updated and revised extensively as and if NCHEMS is able to further develop the model.

4141903000045300:
1C:773:WICHE:PP:2BA90